



**João Miguel
Requeijo Dias**

**Sistema para Pagamento de Portagens
baseado em Smartphone**



**João Miguel
Requeijo Dias**

**Sistema para Pagamento de Portagens
baseado em Smartphone**

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia Electrónica e Telecomunicações, realizada sob a orientação científica do Professor Doutor João Nuno Pimentel da Silva Matos, Professor do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro, e do Professor Doutor Arnaldo Silva Rodrigues de Oliveira, Professor do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro.

o júri / the jury

presidente / president

Professor Doutor José Alberto Gouveia Fonseca

Professor Associado da Universidade de Aveiro

vogais / examiners committee

Professor Doutor Joaquim José de Castro Ferreira

Professor Adjunto da Escola Superior de Tecnologia e Gestão de Águeda

Professor Doutor João Nuno Pimentel da Silva Matos

Professor Associado da Universidade de Aveiro (Orientador)

agradecimentos / acknowledgements

A realização da presente dissertação deve-se à congregação de vários factores, muitos proporcionados por aqueles que me rodeia e sempre apoiaram, aos quais quero agradecer.

Em primeiro lugar, quero agradecer aos meus orientadores pela sua ajuda, coordenação, pelos conhecimentos e espírito crítico transmitidos, confiança e paciência durante este último ano.

A minha enorme gratidão, aos meus pais e ao meu irmão, por todo o apoio, responsabilidade e educação que sempre me inculcaram. Sem eles nada disto era possível.

À Mariline Pinto, pela presença nos momentos chave da minha vida e por toda a atenção, carinho, conselhos e motivação que sempre me transmite.

A todos os meus amigos que me acompanham no dia a dia. Ao lado deles, grandes momentos ficaram para a história, pois com eles a passagem por esta fase da vida foi muito mais divertida. Aos amigos dos Ramos Estudantis do IEEE da Universidade de Aveiro e das outras Universidades, e da AETTUA por tudo aquilo que construímos em conjunto.

Um grande bem haja!

Palavras-chave

Portagens Electrónicas; Sistema Electrónico Europeu de Portagem; GNSS; NFC; OBU multi-tecnológico; C2S; Cobrança de Portagens por *Smartphone*.

Resumo

Desde a década de 60 que a criação de sistemas de pagamento em infra-estruturas rodoviárias tem suscitado interesse entre os governos dos países desenvolvidos, dadas as suas vantagens económicas no que toca à amortização dos investimentos incorridos e à implementação de políticas baseadas no princípio do utilizador pagador. Com a crescente utilização das redes rodoviárias como meio de transporte de mercadorias e pessoas, as questões ambientais, de segurança e de fluidez de trânsito começaram a tornar-se temas relevantes para o desenvolvimento sustentado das sociedades modernas. Assim, o investimento em inovação e tecnologia dos sistemas eletrónicos de pagamento de portagens, tem sido constante nos últimos anos.

Hoje em dia, a maioria dos membros da União Europeia possuem serviços de pagamento de portagens eletrónicas, pois comprovou-se que estes acarretam não só benefícios económicos, mas também benefícios para o cidadão comum, em termos de segurança e qualidade de vida. Contudo, cada país criou os seus sistemas ou serviços, aplicados a circunstâncias específicas e seguindo estratégias díspares, o que levou à proliferação de sistemas de portagem incompatíveis, alguns a nível tecnológico, o que torna a interoperabilidade destes, um desafio atual. O esforço desenvolvido pela Comissão Europeia para superar este desafio levou à criação do Sistema Eletrónico Europeu de Portagem (SEEP).

Na presente dissertação elaborou-se um estudo dos sistemas eletrónicos de pagamento de portagem utilizados na União Europeia, no qual se identificaram algumas limitações para além da falta de interoperabilidade, tal como a ausência de um interface entre o utilizador e os operadores de portagens, que dificultam a utilização dos sistemas eletrónicos de portagem em determinados cenários, como o de *car sharing*, por exemplo.

No sentido de atacar esta questão, desenvolveu-se um sistema denominado Charge Collector System (C2S). Para além da conceptualização da estrutura e do modo de funcionamento, ainda se produziu uma prova de conceito. Esta congrega uma unidade de bordo, ou *On Board Unit* (OBU) baseada em Linux, capaz de comunicar com um *smartphone*, para o qual se criou uma aplicação móvel, baseada em Android. A aplicação disponibiliza um interface gráfico onde o utilizador poderá consultar os registos de portagem incorridos e submetê-los para pagamento. Foram ainda realizados testes de estrada e exibidos os seus resultados.

Os trabalhos desenvolvidos, no âmbito desta dissertação, resultam num modelo ilustrativo, ou prova de conceito, de uma unidade de bordo e de uma aplicação móvel, capaz de expressar toda a experiência de utilização inerente ao funcionamento do sistema C2S, demonstrando o registo de portagens, a posterior transmissão destas para o *smartphone* e a consulta das mesmas em formato de aplicação móvel.

Keywords

Electronic toll collection; EETS; GNSS; multitechnology OBU; NFC; C2S; Smartphone-based Toll Collection System.

Abstract

Since the 60's, the implementation of the toll collection systems has raised interest by the governments of developed countries, due to its economical benefits in terms of the amortization of the incurred investments and the execution of laws based in the user pays principle. The high increasing of road networks, to transport people and goods, put the environmental and security issues, as well as the traffic flow concerns, much more relevant to enhance the sustainable development of modern societies. Thereby, the investment in innovation and technology of electronic toll collection has been a constant on the last years.

Nowadays, most of the European Union members have electronic toll collection on their road networks, because it was proved that its usage leads, not only economical benefits, but also benefits directly to the people, promoting security and quality of life. However each country developed and applied their services or systems in different circumstances, following disparate strategies that led to dissemination of incompatible ETC systems, some of them at the technological level, which makes the implementation of the interoperability feature a real challenge. The European Commission efforts have been developed near the creation of the European Electronic Toll Service (EETS), trying to surpass this pitfall.

On this dissertation, it was developed a study about electronic toll collection used in the European Union, where it was highlighted some of their limitations beyond the interoperability feature, such as the lack of the user interface that difficult the use of ETC services in some scenarios, for instance in the car sharing business.

Moreover, it was developed a system, named Charge Collector Sytem (C2S), where it was elaborated all the structure and the operating mode, as well produced a proof of the concept. This model proof gathers one On Board Unit (OBU) based in Linux systems, able to communicate with a smartphone, for which it was developed an Android mobile application. This mobile application provides a user interface where it is shown the incurred toll logs and where it is possible to send them to a payment system. It was also developed several tests and showed some results.

The project developed under this dissertation resulted in the proof of the concept, constituted by the OBU and the mobile application, where the user can experience the functions of the system C2S, related with the tolling registration, the logs transmission to the smartphone and the logs query on the mobile application.

Conteúdo

Conteúdo	i
Lista de Figuras	iii
Acrónimos	v
1 Introdução	1
1.1 Enquadramento	1
1.2 Motivação	2
1.3 Objetivos	3
1.4 Estrutura da Dissertação	3
2 Estado da Arte	5
2.1 Introdução	5
2.2 Estratégias de aplicação de portagens	6
2.3 Sistemas de Pagamento de Portagem	7
2.3.1 Sistema Electrónico Europeu de Portagem	7
2.4 Evolução de estratégias pela tecnologia	11
2.5 Pontos negativos dos sistemas ORT	12
2.6 Alternativas nacionais para veículos estrangeiros em infraestruturas ORT	12
2.7 <i>Car Sharing</i>	13
2.7.1 <i>Car Sharing vs</i> Sistemas de portagem ORT	15
2.8 Futuros modelos e sistemas: o <i>Congestion Pricing</i>	15
2.9 Soluções no mercado	17
2.9.1 OBUs baseado em GNSS e comunicações móveis GSM – GPRS com módulos DSRC	18
3 <i>Charge Collector System</i>	21
3.1 Introdução	21
3.2 Arquitetura dos Sistemas de Pagamento	23
3.2.1 Arquitetura C2S	24
3.3 Tecnologia	27
3.3.1 Sistemas de navegação por satélite	28
3.3.2 Computador Central	29
3.3.3 <i>Smartphone</i>	31
Android	32
NFC	34

3.4	Modo de utilização	37
3.5	<i>User Cases</i>	38
3.5.1	EETS	38
3.5.2	Aluguer de viaturas ou <i>Car Sharing</i>	39
4	Prova de Conceito	41
4.1	Estrutura e Desenho do Sistema	41
4.1.1	<i>On Board Equipment</i>	41
	Leitura de dados - <i>Sniffer</i>	42
	Comunicação entre processos por meio de <i>Sockets</i>	45
	Processamento de dados - <i>Matcher</i>	46
	Registo e envio de dados - <i>Logger</i>	48
4.2	Aplicação Móvel	52
4.2.1	Implementação das atividades	54
	Tratamento e gestão de registos	56
4.3	Testes e Resultados	59
4.3.1	OBE	59
4.3.2	Aplicação móvel	62
5	Conclusão e Trabalho Futuro	65
5.1	Conclusão	65
5.2	Trabalho Futuro	66
A	Tabela com tecnologias dos países da União Europeia	67
B	Comparação de OBUs com tecnologia GNSS-GSM e DSRC	69
	Bibliografia	71

Lista de Figuras

2.1	Comparação de custos de implementação entre DSRC e GPS-GSM, extraído de [1].	8
2.2	Arquitetura do projeto de interoperabilidade, extraído de [1].	9
2.3	Tecnologia na UE, baseado na apresentação da Kapsch em Sofia, Bulgária. . .	10
2.4	Pórtico de uma infraestrutura ORT.	11
2.5	Toll Card.	13
2.6	<i>Car Sharing</i> [2].	14
2.7	Aplicação móvel <i>GetAround</i> - consiste numa espécie de rede social onde os seus utilizadores podem partilhar e alugar os seus veículos [3].	14
2.8	Modelo do <i>Congestion Pricing</i> <i>Congestion Pricing</i>	16
2.9	a) OBU 1372 – <i>Dashboard Solution</i> ; b) OBU 1373 – <i>DIN Slot Solution</i> ; c) OBU 1374 – <i>Windscreen Solution</i> ; d) OBU 1376 – <i>The Swiss LSV</i> OBU [4].	18
2.10	TriponCH.	19
2.11	TRIPONEU.	19
2.12	TRIPONSM.	20
2.13	hybridOBU	20
3.1	Ação de pagamento multi utilizador/pagador com Proprietário definido por defeito.	22
3.2	Arquitetura do sistema MLFF da Ascendi. Baseado e extraído de [5]	23
3.3	Estrutura global de operação do C2S e das restantes entidades.	25
3.4	Protocolo de comunicação entre OBE, a aplicação móvel e o <i>back-office</i>	26
3.5	Proposta de modelo para atualização das base de dados.	26
3.6	Estrutura de uma mensagem NDEF.	36
3.7	Estrutura de um registo NDEF.	36
3.8	Modo de funcionamento do C2S, passo a passo, numa infraestrutura DSRC. .	37
4.1	Estrutura organizacional do <i>On Board Equipment</i>	42
4.2	Fluxograma da rotina do módulo <i>Sniffer</i>	44
4.3	Fluxograma da criação de um <i>socket</i> , baseado em [6].	45
4.4	Falha de detecção em curvas de transição: a) com ausência de erro; b) com erro na longitude.	47
4.5	Proposta de localização de zonas de portagem: a) uma única zona; b) zonas desfasadas.	48
4.6	Fluxograma da rotina do módulo <i>Matcher</i>	49
4.7	Fluxograma da rotina do módulo <i>Logger</i>	51

4.8	Fluxograma geral do OBE.	52
4.9	Algumas imagens do <i>mockup</i> produzido para o desenho da aplicação C2S. . .	53
4.10	Estrutura organizacional do ciclo de vida da atividade principal ou <i>main</i> . . .	55
4.11	Estrutura organizacional do ciclo de vida da atividade "toll_selection". . . .	57
4.12	Ciclo de um registo na segunda atividade, "toll_selection".	58
4.13	Trajeto definido e resultado do <i>software</i> SatGenNMEA.	60
4.14	Zonas de portagem do primeiro ensaio do teste de estrada.	61
4.15	Zonas de portagem do segundo ensaio do teste de estrada, correspondentes aos pórticos existentes.	62
4.16	Algumas imagens da aplicação móvel C2S.	62

Acrónimos

ALPR Automatic License-Plate Recognition v, 11, 24, 27, 37–39

API Application Programming Interface v, 24, 59, 65, 66

ARMHF ARM hard float v, 30, 31

C2S Charge Collector Sytem v, 3, 21, 22, 24, 27, 28, 32, 34, 37–39, 41, 53

CBO Comercial Back-Office v, 24

daemon Disk And Execution MONitor v, 43

DSRC Dedicated Short Range Communications v, 2, 6–11, 18, 21–24, 27, 28, 37–39, 51, 52, 65

EETS European Electronic Toll Service v, 2, 3, 7–9, 11, 17, 21, 27, 38, 39, 41, 65

EGNOS European Geostationary Navigation Overlay Service v, 29, 43

ETC Electronic Toll Collection v, 1, 3, 5, 6, 11, 38

GLONASS Globalnaya Navigatsionnaya Sputnikovaya Sistema v, 29

GNSS Global Navigation Satellite System v, 2, 6, 9, 21, 22, 24, 27, 37, 38

GPIO General Purpose Input/Output v, 28, 30, 48, 52

GPRS General Packet Radio Service v, 7, 65

GPS Global Positioning System v, 7–10, 18, 24, 27–29, 32, 41–43, 45–47, 59, 60, 65, 66

GSM Global System for Mobile Communications v, 2, 6–10, 18, 21, 22, 24, 27, 38

I2C Inter-Integrated Circuit v, 30

IMTT Instituto da Mobilidade e dos Transportes Terrestres v, 24

ITS Intelligent Transportation Systems v, 29

LED Light Emitting Diode v

MLFF Multi-Lane Free-Flow v, 23

NDEF NFC Data Exchange Format v, 35, 55

NFC Near Field Communication v, 17, 22, 28, 34–36, 39, 41, 42, 48–51, 53–55, 58, 59, 62, 65, 66

NTP Network Time Protocol v, 30

OBE On Board Equipment v, 3, 24–29, 31, 34, 35, 37, 38, 41, 42, 45, 51–55, 58–60, 62, 63, 65, 66

OBO Operational Back-Office v, 24

OBU On Board Unit v, 1, 2, 7–9, 11, 12, 17, 18, 21, 22, 24, 37–39, 52

ORT Open Road Tolling v, 1, 6, 7, 12, 14, 21, 39, 41

P2P Peer-to-Peer v, 48

PDA Personal Digital Assistant v, 32

RFID Radio-Frequency IDentification v, 34

RPi Raspberry Pi v, 29–31, 42, 43, 49, 52, 59

RSE Roadside Equipment v, 9, 24

RSU Road Side Unit v, 7

RTC Real Time Clock v, 30

SCP Secure Copy Protocol v, 59

SDK Software Development Kit v, 32, 33

SEEP Sistema Electrónico Europeu de Portagem v, 2

SIM Subscriber Identity Module v, 17

SoC System on Chip v, 29

SPI Serial Peripheral Interface v, 30, 49

SSH Secure Shell v, 30, 59

TNF Type Name Format v, 35

UART Universal Asynchronous Receiver/Transmitter v, 49

USB Universal Serial Bus v, 28–30, 34, 42

WAAS Wide Area Augmentation System v, 43

Capítulo 1

Introdução

1.1 Enquadramento

Desde a década de 60 que a criação de sistemas de pagamento em infraestruturas rodoviárias tem suscitado interesse por parte dos governos dos países desenvolvidos [7], dadas as suas vantagens económicas no que toca à amortização dos investimentos incorridos, à possibilidade de construir ou remodelar novas infraestruturas em menores espaços de tempo e à implementação de políticas baseadas no princípio do utilizador pagador, entendidas como mais justas. Com a crescente utilização das redes rodoviárias como meio de transporte de mercadorias e pessoas, as questões ambientais, de segurança e de fluidez do trânsito começaram a tornar-se tema de discussão, sendo definidos princípios preponderantes para o desenvolvimento sustentado das sociedades modernas. Assim, o investimento em ciência e tecnologia, para a criação de soluções que defendam os princípios referidos, tem sido intenso e a inovação nos sistemas de pagamento de portagens é uma constante bem patente nas últimas décadas.

Hoje em dia, a maioria dos membros da União Europeia possuem serviços de pagamento de portagens eletrónicas, ou Electronic Toll Collection (ETC), pois comprovou-se que estes acarretam benefícios para ambas as partes intervenientes. Os Estados têm benefícios económicos como já referido e os cidadãos utilizadores, graças à componente tecnológica, saem beneficiados com o facto de estes serviços permitirem o pagamento em movimento, diminuindo o risco de acidente e o congestionamento de trânsito, que consequentemente geram benefícios secundários, como a diminuição de emissão de gases poluentes por exemplo. O sucesso destes serviços é bem evidente, bastando observar a evolução que estes têm demonstrado nos últimos tempos com a criação de sistemas de pagamento de portagens totalmente eletrónicos, também conhecidos por Open Road Tolling (ORT), onde a famosa praça de portagem é substituída por um pórtico e o utilizador não tem a opção de pagamento manual. Contudo, nos últimos anos, cada país tem desenvolvido os seus sistemas ou serviços, aplicados a circunstâncias específicas e seguindo estratégias díspares, o que levou à proliferação de sistemas de pagamento de portagem incompatíveis, alguns a nível tecnológico, o que torna a aplicação do princípio da interoperabilidade um desafio atual, no qual os desenvolvedores e produtores de sistemas devem-se debruçar.

Os esforços desenvolvidos para a criação de um sistema global de pagamento de portagens, que permita que várias entidades interoperem e que os condutores viajem livremente pela Europa, com uma única unidade de bordo ou On Board Unit (OBU), têm sido constantes por parte da Comissão Europeia. Desta forma criou-se o projeto Europeu para o desenvolvimento

do Sistema Electrónico Europeu de Portagem (SEEP), ou European Electronic Toll Service (EETS), o qual já emitiu uma diretiva, 2004/52/CE [8], que define o tipo de organização que o serviço terá, as entidades intervenientes e o tipo de tecnologia necessária. O projeto mantém-se em desenvolvimento embora ainda não esteja concluído, sendo no entanto possível ter um OBU a operar em sistemas baseados em diferentes tecnologias, Dedicated Short Range Communications (DSRC) ou Global Navigation Satellite System (GNSS) com Global System for Mobile Communications (GSM), apesar de a componente processual, isto é, burocrática e negocial não ser simples, envolvendo um grande esforço das entidades intervenientes.

A ausência da interoperabilidade entre sistemas de pagamentos de portagem é uma funcionalidade bastante perceptiva pelos utilizadores e pelos desenvolvedores dos sistemas, no entanto existem outras falhas, ou ausências de funcionalidades nos sistemas atuais, que em conjunto com a interoperabilidade poderiam proporcionar soluções mais amigas do utilizador.

O presente trabalho baseia-se na proposta de uma solução para as falhas descrita e os seus princípios fundamentais e o desenho da solução, foram aprovados e apresentados sob artigo científico na conferência VEHICULAR 2013 em Julho de 2013, em Nice, França. Existe ainda um segundo artigo desenvolvido no âmbito da dissertação, já com a implementação final que foi aceite para a conferência CETC 2013 (Conference on Electronics, Telecommunications and Computers), na qual vai ser apresentado em Dezembro de 2013, em Lisboa.

1.2 Motivação

Os sistemas atuais estão tecnologicamente baseados, na sua grande maioria, em comunicações DSRC e ainda em menor número em GNSS/GSM. Os primeiros são mais utilizados dadas as suas boas performances a nível de segurança e a sua robustez em mecanismos de *enforcement*, ou seja, fiscalização e auditoria, além disso, historicamente foram desenvolvidos primeiro. Os segundos são utilizados em cenários de aplicação de portagem a veículos pesados de transporte de mercadorias, onde a questão da privacidade do cidadão comum não se coloca e procura-se portajar grandes extensões rodoviárias com um investimento reduzido. Atualmente, os casos de interoperabilidade entre sistemas desta natureza são raros. Embora no passado, a razão para a não inter-operação estar relacionada com a limitação tecnológica dos OBUs, hoje em dia já existem diversos fabricantes a produzir tecnologia preparada para este cenário, como exposto no capítulo Estado da Arte.

Com este facto, infere-se facilmente que os factores para a implementação da interoperabilidade dos sistemas na Europa, estão para lá do OBU. Tal como previsto na diretiva Europeia para a implementação do EETS, as estruturas instaladas nos sistemas atuais terão de se flexibilizar e comunicar com outras entidades, entidades Europeias por exemplo.

Atualmente, os serviços de pagamento de portagem em operação, ficam aquém de um conjunto de funcionalidades que englobam a interoperabilidade e a flexibilidade de utilização. As funcionalidades em falta, passam por um interface gráfico, de fácil acesso, que permita a consulta e o pagamento das portagens incorridas, bem como a definição de diversas opções como o modo de pagamento, por exemplo. Por outro lado, o facto de existir um interface entre o utilizador e a entidade de cobrança, permite uma comunicação bidirecional que poderá dar azo à implementação de taxas de portagem dinâmicas, importantes em certos cenários como o de entrada e saída do centro de cidades.

A extensão das funcionalidades apresentadas em relação à da interoperabilidade, surge do facto de o número de utilizadores de veículos, dos quais não são proprietários, estar a

aumentar, como é o caso do aluguer de viaturas ou sistemas de *car sharing*. Nestes casos a existência de uma solução que permitisse a aferição das portagens incorridas pelo utilizador e respectivo pagamento imediato, facilitaria o modo de utilização e diminuiria os encargos administrativos das empresas de aluguer. Estes factos, aliados à presente remodelação dos sistemas de pagamento para a inserção da interoperabilidade como funcionalidade, constituem a motivação para o desenvolvimento da presente dissertação.

1.3 Objetivos

Os trabalhos desenvolvidos no âmbito da presente dissertação têm como objectivo geral criar uma solução que proporcione uma experiência de utilização mais simples e cómoda, dos sistemas eletrónicos de pagamento de portagens. Este objectivo geral foi particionado em etapas/objetivos intermédios de forma a se criar uma linha geral condutora do projeto. Os objetivos definidos foram:

- Estudar e descrever o estado da arte, onde conste a tecnologia existente, as soluções presentes no mercado, os sistemas inovadores, bem como as estratégias desenvolvidas e necessidades dos utilizadores dos sistemas eletrónicos de pagamento de portagem;
- Compreender o EETS, a sua estrutura, implementação e o estado atual do projeto;
- Desenhar um modelo de sistema que reúna funcionalidades que constituam soluções para os utilizadores dos sistemas eletrónicos de pagamento de portagem;
- Desenvolver uma prova de conceito que exemplifique o modelo do sistema desenhado;
- Elaborar testes que validem a prova de conceito apresentada.

1.4 Estrutura da Dissertação

A presente dissertação divide-se em 5 capítulos:

Capítulo 1 - Introdução é feito um breve enquadramento do tema em causa. É onde são referidos os objetivos e motivações que o guiam e relatada a estrutura da dissertação de modo a permitir uma melhor percepção do conteúdo referente a cada capítulo.

Capítulo 2 - Estado da arte onde é explicado o motivo pelo qual existem diferentes sistemas eletrónicos de pagamento de portagem, onde são expostos os sistemas existentes na União Europeia, as suas características e tecnologias, e o projeto EETS. É ainda demonstrada a evolução tecnológica existente nestes sistemas, as suas falhas para o utilizador, as que mais prejudicam e as alternativas existentes. O capítulo encerra com as soluções existentes no mercado, apresentadas por diversos fabricantes.

Capítulo 3 - Charge Collector System é o sistema desenhado para fornecer funcionalidades de interface gráfico, onde o utilizador poderá consultar os registos de portagem e submetê-los para pagamento. O Charge Collector Sytem (C2S) utiliza ambas as tecnologias características dos ETC e pretende-se um sistema interoperável. Neste capítulo é descrito o modo como o C2S foi desenhado, os módulos que o constituem, bem como estes se articulam e operam. Ainda é explicado que tipo de tecnologias foram utilizadas para a sua implementação e o modo de utilização do mesmo, através de dois *user cases*.

Capítulo 4 - Prova de Conceito onde é explicada a estrutura de operação do sistema C2S e a implementação dos diversos módulos (o On Board Equipment (OBE) e a aplicação móvel) da prova de conceito. São ainda apresentados os resultados e as funcionalidades implementadas bem como o tipo de testes a que se sujeitou o sistema.

Capítulo 5 - Conclusão e Trabalho Futuro onde é avaliado o trabalho desenvolvido tendo em conta os objetivos definidos, são revelados os principais desafios encontrados e são propostos novos desenvolvimentos no trabalho, com vista a criar alicerces para concepção de um futuro protótipo.

Capítulo 2

Estado da Arte

2.1 Introdução

Os sistemas de portagens eletrónicas, ou ETC, são bastante utilizados por todo mundo graças às vantagens proporcionadas pelo tipo de utilização a adoptar por parte dos condutores, apoiada por uma componente tecnológica simples e eficiente, resultando em:

- Benefícios para a segurança rodoviária [7], possibilitando um fluxo regular de trânsito e evitando paragens, os sistemas ETC diminuem, ou eliminam, o perigo de acidente devido a reduções ou paragens bruscas dos veículos;
- Benefícios económicos [9], aplicando as receitas provenientes das taxas de portagem à amortização do investimento inicialmente realizado para a construção das infraestruturas, bem como nos trabalhos de manutenção. Por outro lado, reduzindo a afluência de tráfego em grandes cidades no transporte de mercadorias ou na viagem de passageiros, diminuindo assim atrasos e consecutivamente custos de transporte, desequilíbrios de *stock* e custos/prejuízos no consumidor final. Além disso, ao proporcionar passagens mais rápidas e menos paragens, os sistemas ETC promovem uma distribuição de tráfego pela rede rodoviária mais eficiente, equilibrando a utilização e o desgaste entre os locais de portagem e os restantes da infraestrutura;
- Benefícios ambientais [7], diminuindo o congestionamento de tráfego, resulta numa direta redução das emissões de gases e poluição sonora.

No entanto é de assinalar que tais benefícios só serão sentidos em circunstâncias de utilização massiva por parte dos utilizadores de portagens e para isso é necessário ressaltar certas condições práticas [10], tais como:

- Satisfazer o desejo, legítimo dos condutores, de beneficiar de uma melhor qualidade de serviço nas infraestruturas rodoviárias relativamente às alternativas existentes;
- Facilitar a utilização e a forma de pagamento;
- Tratar os dados pessoais dos utilizadores respeitando a legislação em vigor quanto à proteção e à livre circulação de informação no sector das comunicações eletrónicas;
- Criar planos de isenção e descontos a atribuir de forma flexível e rápida, protegendo os mais desfavorecidos;

- Evidenciar as vantagens do utilizador, quanto ao preço de portagem a pagar comparado com o serviço a desfrutar;

Portanto, todo o tipo de medidas tomadas pelos governos ou entidades responsáveis devem assentar numa forte intensificação na utilização de portagens eletrónicas para que os benefícios relatados possam ser constatados [11].

Na última década, devido ao sucesso registado pelos sistemas ETC, com o nível de adesão de novos utilizadores e da resposta tecnológica exigida, começaram a surgir soluções de pagamento de portagem totalmente, ou exclusivamente, eletrónicas, denominadas ORT [12], baseando-se em tecnologias DSRC ou GNSS com GSM. As vantagens inerentes aos sistemas ORT, estão ligadas à redução de custos operacionais por parte das entidades concessionárias e a flexibilização (instalação, utilização e atualização) dos sistemas.

No entanto existem vulnerabilidades e defeitos apontados aos sistemas ORT e na generalidade aos ETC. A principal falha está na inexistência, ou na inoperação, dos mecanismos de interoperabilidade entre sistemas vizinhos. Por outro lado há limitações nos sistemas existentes, principalmente nos ORT, no que toca ao acesso à informação relacionada com as portagens incorridas pelos utilizadores e às opções ou formas de pagamento. Estas falhas são mais notáveis no âmbito Europeu e ao longo do presente capítulo serão apresentadas as razões para a difusão destes sistemas de pagamento de portagens heterogéneos, ou seja as estratégias utilizadas por cada entidade para "portajar" determinada infraestrutura. Seguidamente, serão apresentadas as tecnologias e soluções utilizadas atualmente, as suas limitações e benefícios, bem como o futuro dos modelos de portagens baseado no *Congestion Pricing* e é ainda discutida a situação atual dos sistemas de pagamento de portagens nacionais portuguesas.

2.2 Estratégias de aplicação de portagens

A implementação e aplicação dos sistemas de pagamento de portagens, pode ser orientada por diferentes estratégias. Segundo a consultora Deloitte [7], cada estratégia surgiu para diferentes propósitos consoante a infraestrutura onde se pretende aplicar, possuindo níveis de complexidade díspares e consequentemente diferentes tipos de tecnologia.

Existem várias estratégias já implementadas:

- Na secção de estrada: portagens mais utilizadas, implementadas em pontes, túneis, vias rápidas, etc., nas quais o utilizador paga uma taxa de utilização por um itinerário de estrada ou ponte. O objectivo principal da estratégia, onde se incluem este tipo de portagens, é financiar o custo de construção e de operação das respectivas infraestruturas. A complexidade tecnológica associada a este sistema pode ser tão baixa quanto possível (portagens com pórticos manuais). Associado a este facto surge a grande desvantagem de este tipo de cobrança gerar bastante congestionamento de tráfego;
- Na área ou zona: portagens aplicadas em áreas específicas, por exemplo centros de cidades. O objectivo principal é reduzir o número de veículos e consequentemente os níveis de poluição em determinadas áreas. A complexidade tecnológica exigida por esta estratégia é mais elevada, visto que cada ponto de entrada/saída deve identificar os veículos já cobrados de forma rápida e eficaz por forma a evitar congestionamentos. É ainda importante referir que para o cumprimento dos objetivos da referida estratégia é necessário incluir nesta alternativas de transporte viáveis para os utilizadores, como

por exemplo uma rede de transportes públicos articulada com horários de maior tráfego e com preços competitivos;

- Na distância percorrida: estratégia muito pouco utilizada, mas tecnologicamente já implementada, onde os veículos são cobrados com base numa taxa de utilização de estrada (*vehicle-miles-traveled*). O objectivo principal é gerar receitas para o sistema de transportes de forma a regular a quantidade de veículos e os efeitos negativos associados. A complexidade deste sistema é relativamente alta e exige a aplicação de equipamentos de medição, localização e comunicação móveis em todos os utilizadores. Outros aspectos problemáticos surgem relacionados com o *enforcement*, ou seja, fiscalização e garantia de pagamento das taxas, e com a privacidade dos utilizadores.

2.3 Sistemas de Pagamento de Portagem

Tal como referido anteriormente, a existência de diferentes cenários e diferentes estratégias, levou a que cada país agi-se como uma entidade individual e escolhe-se um sistema tecnologicamente mais adequado às suas circunstâncias.

O sistema mais utilizado em todo mundo tem uma estrutura bastante semelhante nas suas aplicações [1]. Este baseia-se na comunicação entre *tags* eletrónicas ou o OBU, instaladas no interior dos veículos, e os pórticos ou Road Side Unit (RSU) equipados com comunicações microondas 5,8 GHz, chamado DSRC, sistemas de fiscalização (câmara de vídeo, por exemplo), os ditos *enforcements* e sensores de classificação de veículo. Uma infraestrutura (estrada, por exemplo) tem vários RSUs, para controlar entradas e saídas, conectados entre si e com o sistema de *back-office* que se encontra ligado ao departamento responsável pelo processamento da faturação. Em paralelo com os RSUs poderão existir sistemas manuais ou semi-automáticos (exigem paragem e interação do utilizador) de pagamento, mas no caso de infraestrutura ORT, existe apenas o RSU com pagamento automático e o sistema na sua globalidade (*enforcement* e classificação de veículo) é mais autónomo. Outro sistema bastante utilizado é baseado em telecomunicações móveis GSM e num sistema de posicionamento global baseado em satélite, como o Global Positioning System (GPS) [1], onde o OBU é capaz de calcular a distância percorrida pelo veículo, através do GPS e outros sensores de posicionamento, bem como a "portagem" que o utilizador deve pagar, transmitindo esta para o *back-office* onde acaba de ser processada e debitada à conta do cliente.

Na União Europeia os sistemas não são interoperáveis, devido a diferentes fatores como referido anteriormente, mas o principal é a estrutura tecnológica que se encontra inerente a cada um. Portanto a Comissão Europeia iniciou o projeto do Sistema Eletrónico Europeu de Portagem ou EETS, com a aprovação da Diretiva 2004/52/EC a 29 de Abril 2004 pelo Parlamento Europeu. Por outro lado, a União Europeia deposita bastante confiança no investimento realizado no sistema Galileo, como melhoramento tecnológico em relação ao GPS.

2.3.1 Sistema Electrónico Europeu de Portagem

A interoperabilidade entre sistemas de pagamento de portagens, na União Europeia, é um objectivo há muito procurado, difícil de implementar e por isso considerado um objectivo a longo prazo. A criação do projeto do EETS ambiciona uniformização da tecnologia dos sistemas, utilizando navegação por satélite, comunicações móveis GSM ou General Packet

Radio Service (GPRS) (standard GSM TS 03.60/23.060) e comunicações microondas (DSRC), através do desenvolvimento de trabalhos por grupos técnicos, por forma a se chegar a um consenso. Mais do que a componente técnica, existem ainda as componente processual e contratual no qual os grupos de trabalho tiveram de se debruçar, desde 2004. A primeira meta do projeto EETS remete para a data de 1 de julho de 2011, a partir da qual as instituições devem implementar sistemas eletrónicos de portagem.

Tal como referido anteriormente, a principal componente na interoperabilidade prometida, reside no nível técnico. A discussão estabelece-se entre as duas tecnologia utilizadas, qual a mais indicada e viável para os sistemas de portagem. Segundo a Austríaca EFKON AG [13], os custos de implementação de sistemas GPS-GSM são um pouco mais elevados (cerca de 20%) do que de sistemas DSRC, para infraestruturas com menos de 1000 km e com 300000 OBUs. A partir de 1000 km e para o mesmo número de OBUs, os custos de implementação do GPS-GSM decaem, para 60% no caso de 3000 km de estrada por exemplo, enquanto que para os mesmos valores os custos do sistema DSRC decaem ligeiramente, tal como ilustra a Figura 2.1. Portanto os sistemas GPS-GSM são indicados para infraestruturas de larga escala, ou seja, entidades concessionárias de diferentes autoestradas por exemplo. Embora este sistema tenha custos de implementação mais baixos, tem a desvantagem dos mecanismos de *enforcement* serem mais fracos e portanto mais sujeitos à violação. Por outro lado, coloca-se a questão da privacidade da partilha de localização dos utilizadores, visto que o sistema pode comunicar constantemente, podendo o veículo ser monitorizado remotamente. Neste momento os sistemas GPS-GSM são maioritariamente utilizados para veículos pesados (mais do que 12 toneladas) para transporte de mercadorias, como é o caso da Alemanha, com o sistema a ter início de operação em 2005, contando com 12500 km de autoestradas portajadas, mais de 18,5 milhões de transações por dia e receitas a rondar os 4,5 biliões de Euro em 2010 [14].

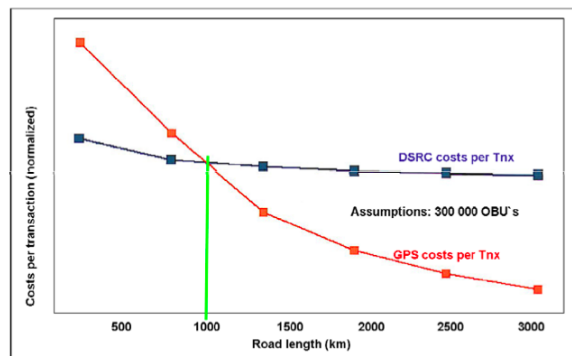


Figura 2.1: Comparação de custos de implementação entre DSRC e GPS-GSM, extraído de [1].

Num consórcio de 27 entidades, no qual participaram operadores, concessionárias, produtores de tecnologia, etc., a Comissão Europeia desenvolveu um projeto de implementação e teste das portagens interoperáveis, entre 2005 e 2008, em seis países vizinhos: Áustria (com a ASFINAG), Alemanha (TOLL COLLECT), Itália (TELEPASS), França (TIS PL), Espanha (VIA-T) e Suíça (LSVA). Este projeto teve como objectivo demonstrar e validar protótipos desenvolvidos até então, sob os *standards* produzidos pelos grupos de trabalho e pelo *European Committee for Standardization* (CEN). Ao mesmo tempo foram definidos alguns aspetos durante o projeto, tal como um nível superior da arquitetura a utilizar para o EETS, ainda

não definido até à data. Esta arquitetura estabelece os interfaces necessários entre as entidades concessionárias ou *Toll Chargers*, a entidade agregadora e fornecedora do serviço ou *Toll Service Provider* e o utilizador ou *Service User*, tal como ilustra a Figura 2.2. Apresentada em 2007, esta arquitetura tem como princípio a cobrança em tempo real através da comunicação DSRC entre o OBU e a infraestrutura (Roadside Equipment (RSE)), ou o processamento das portagens no OBU recorrendo ao GNSS, sendo posteriormente enviadas para *back-office* do *EETS provider* via GSM e só depois enviado para o *Toll Charger*. Este pode ainda recorrer ao sistema DSRC para aplicar mecanismos de *enforcement*.

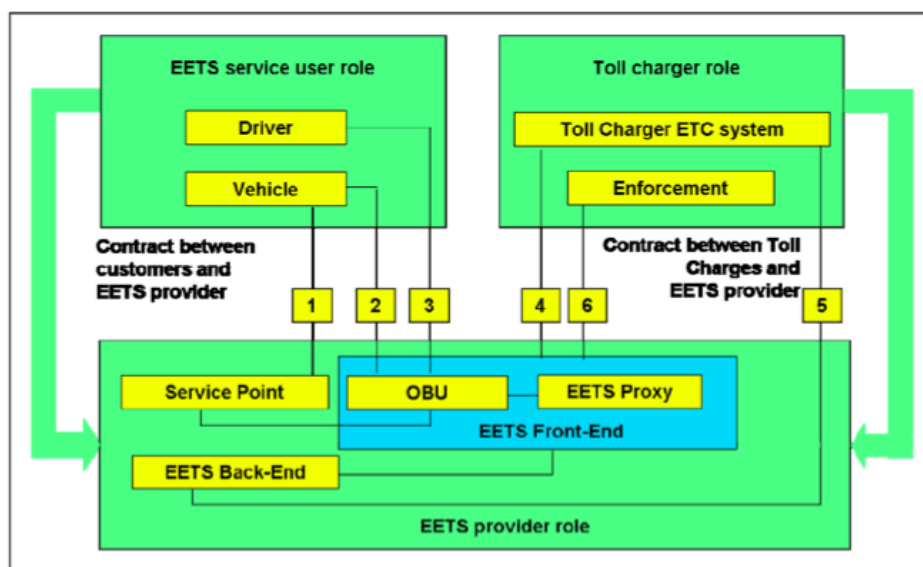


Figura 2.2: Arquitetura do projeto de interoperabilidade, extraído de [1].

Tal como se observa na Figura 2.2, existem vários interfaces para realizar comunicações entre os blocos estruturais, servindo para acesso e manutenção do OBU (1), criação de aplicações adicionais relevantes para partilha de dados como por exemplo navegação GPS (2), interface de utilizador (3), envio de dados de utilização, ou as portagens propriamente dito, do EETS *front-end* para o *Toll Charger back-end* (4), configuração e parametrização da informação a ser enviada (5), e utilização de mecanismos de *enforcement* e auditoria das transações do OBU.

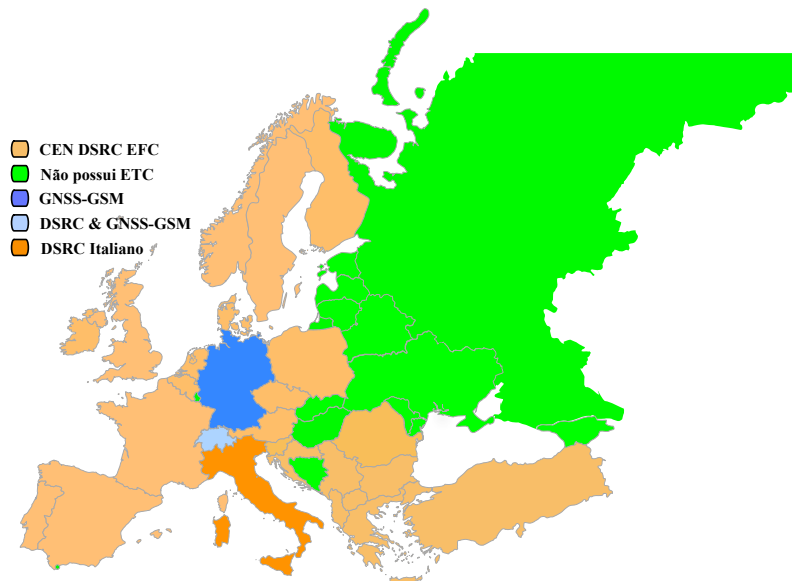
Depois da concepção e implementação da arquitetura, seguiu-se a fase de demonstração onde equiparam dois veículos (pesados), com dois OBUs interoperáveis (um dos quais o TRIPON EU da Figura 2.11), e estes operaram autonomamente, sem intervenção humana, em Espanha, França, Itália, Suíça, Alemanha e Áustria. No final desta fase conclui-se que se deveria continuar os trabalhos de normalização da tecnologia e trabalhar sobre a componente contratual, definir e finalizar a arquitetura técnica do EETS, preparar protótipos e projetos piloto, os OBUs a utilizar deverão possuir módulo de comunicação DSRC, GSM, navegação por satélite (receptor GPS), algoritmos de posicionamento, de forma a garantir a interoperabilidade tecnológica, entre outras conclusões.

Com o projeto referido, o EETS consolidou-se e tomaram-se passos importantes para a interoperabilidade. As tecnologias aplicadas foram baseadas nas mais utilizadas pela União Europeia, como se pode observar pela Figura 2.3. No entanto, a Diretiva 2004/52/CE acrescenta

que é permitido utilizar outro tipo de tecnologias [8], desde que estas respeitem as normas técnicas e não impliquem um ónus adicional para os utentes nem criem discriminações entre eles. Atualmente existem outras propostas alternativas baseadas em odometria ou células de comunicações móveis [15].



(a) Membros da União Europeia [16]



(b) Distribuição de Tecnologias

Figura 2.3: Tecnologia na UE, baseado na apresentação da Kapsch em Sofia, Bulgária.

Facilmente se observa que a maior parte dos países da União Europeia utiliza o sistema CEN DSRC EFC, existindo contudo algumas exceções, como o Telepass 5.8Ghz DSRC de Itália que utiliza uma *tag* DSRC diferente do CEN DSRC, e a Alemanha e Suíça que utilizam o sistema GPS-GSM para veículos pesados. Cada caso pode ser observado mais em pormenor

no Anexo A.

Com o EETS pretende-se definir uma nova classe de sistemas ETC, baseado nas duas tecnologias existentes. Tirando partido da flexibilidade e cobertura das comunicações móveis e ao mesmo tempo da navegação por satélite para a poupança de infraestruturas de estrada pesadas e caras. Por outro lado, garantir suporte das comunicações microondas (DSRC) para se manter uma continuidade e sustentabilidade das infraestruturas já existentes, reaproveitando estas para mecanismos de *enforcement* se necessário. Finalmente, procura-se centrar o processamento das portagens incorridas o quanto mais possível no OBU, de maneira a que este calcule distâncias percorridas, quantidade emissões de gases poluentes, etc., seja flexível e fácil de otimizar, e possua suporte para novas aplicações, como software de gestão de veículo, conexão a serviços de emergência, monitorização para seguros *pay as you drive*, etc.

A aplicação de portagens constitui um método justo de aquisição de receitas para suportar custos de infraestruturas, já que a alternativa aplicada em alguns países, é criar um imposto extra ou aumentar impostos de circulação ou de combustíveis, prejudicando aqueles que não utilizam infraestruturas portajadas. Portanto as portagens têm uma base conceptual sólida, onde é possível evoluir estrategicamente, ou seja no modo de utilização e pagamento, utilizando cada vez mais a tecnologia para fomentar essa evolução.

2.4 Evolução de estratégias pela tecnologia

As estratégias apresentadas anteriormente evoluíram ao longo da história em paralelo com a tecnologia, no sentido de colmatar os pontos negativos de cada uma. O maior avanço registado foi a adopção do ETC. Graças ao sucesso do desempenho do ETC e ao avanço da tecnologia, as concessionárias e as empresas ligadas ao sistema de transporte rodoviário decidiram apostar em estratégias totalmente eletrónicas com *free-flow* total, onde o utilizador não tem a opção de parar para pagar manualmente. Este modelo designa-se de *Open Road Tolling* (ORT), Figura 2.4 e um exemplo nacional são as ex-SCUT.



Figura 2.4: Pórtico de uma infraestrutura ORT.

Num sistema ORT o modo de funcionamento, por defeito parte do princípio que todos os utilizadores possuem um OBU. No caso de um veículo não estar equipado com um OBU e este passar num pórtico, é tirada uma foto à matrícula para identificação do mesmo via Automatic License-Plate Recognition (ALPR). Posteriormente a portagem está disponível para pagamento e é suposto o utilizador dirigir-se a um balcão autorizado, correios ou *payshops* por exemplo, a fim de pagar as portagens, muitas vezes com uma taxa de acréscimo por não possuir OBU. Ao fim de um prazo legal, caso as portagens não tenham sido liquidadas, o procedimento passará por enviar uma notificação para a morada associada à matrícula do veículo.

Os sistemas ORT representam uma intensificação das vantagens dos sistemas ETC tradicionais. Estes têm especial importância nas portagens baseadas na área ou zona, em centros de cidades onde o congestionamento de tráfego pode ser crítico. Basicamente, num sistema desta natureza, o condutor não tem de se distrair com mais nada senão conduzir.

2.5 Pontos negativos dos sistemas ORT

Contudo, o ORT não responde a todos os pedidos dos utilizadores, este herda alguns dos pontos negativos do ETC, como a falta de interoperabilidade entre sistemas vizinhos, com a agravante de não possuírem sistemas de pagamento manuais junto aos pórticos para servir como alternativa aos utilizadores que não possuam OBU ou sejam estrangeiros.

Outro ponto negativo surge do facto da disponibilização das facturas para pagamento ser morosa, podendo levar dias, o que levanta problemas para utilizadores estrangeiros que alugam viaturas. Para melhor exposição dos factos, segue-se um exemplo prático:

- Problema: no caso hipotético de um visitante estrangeiro que alugue um veículo, este fornece os dados do seu cartão de crédito à companhia de aluguer para que possíveis custos adicionais relativos ao período de aluguer possam ser debitados, como por exemplo passagem em portagens ORT. Muitas vezes estes débitos surgem passadas semanas nas contas dos clientes, visto que as faturas só são processadas posteriormente, sendo mais demorado ainda no caso dos veículos não possuírem OBU e as companhias de aluguer terem de se dirigir a balcões autorizados para aceder às faturas.
- Resultado: muitas vezes os clientes ficam desagrados com os valores debitados pois não associam ao número de passagens em portagens que realizaram. Mais, as companhias de aluguer têm custos operacionais e administrativos na consulta/deslocação a balcões autorizados para pagamento das portagens. Embora existam sistemas que já disponibilizem as faturas eletronicamente, estas estão sempre sujeitas a atrasos associados ao processamento de *back-office*, sempre necessário. Por outro lado, no caso de modelos de aluguer de curta duração, os ditos *car sharing*, onde é essencial o utilizador ficar identificado e criar condições/oportunidades de pagamento no momento, surgem problemas relacionados com a atribuição das passagens aos períodos de aluguer.

Este tipo de problemas são bastante frequentes, no entanto é a falta de interoperabilidade entre sistemas que lesa mais os utilizadores normais e empresas com necessidade de viajar frequentemente para o estrangeiro. Embora estejam a desenvolver-se algumas soluções, ainda não existe uma solução global. Alguns países têm apresentado alternativas para que os veículos estrangeiros possam realizar pagamentos em ORTs. Vejamos o caso do sistema Português.

2.6 Alternativas nacionais para veículos estrangeiros em infraestruturas ORT

Portugal foi dos primeiros países a apresentar soluções para veículos de matrícula estrangeira que visitem o país, muito provavelmente devido à forte presença de sistemas ORT nos principais troços de estrada que ligam o país a Espanha. Existem 4 sistemas que oferecem soluções a veículos de matrícula estrangeira:

- Toll Card – consiste num cartão associado à matrícula da viatura e pré-carregado com valores de 5, 10, 20 ou 40 euros, que são consumidos em função da utilização nas portagens exclusivamente eletrónicas. Após aquisição do cartão, este é ativado através de SMS por telemóvel com o código impresso no cartão e a matrícula da viatura para que o mesmo seja associado ao veículo. Tem a validade de 1 ano ou até esgotar o saldo. Este pode ser adquirido presencialmente nos *Welcome points* da Easy Toll ou nos sites dos CTT e da *Portugal Tolls*;



Figura 2.5: Toll Card.

- Toll Service - o condutor poderá adquirir um título pré-pago, pronto a utilizar, com um custo fixo de 20 euros, com utilização ilimitada durante 3 dias. Este sistema só é válido para veículos ligeiros. Este pode ser adquirido presencialmente nos *Welcome points* da Easy Toll ou através dos sites dos CTT ou da *Portugal Tolls*;
- Easy Toll - é uma solução de pagamento automático de portagens eletrónicas com a associação de um cartão bancário (Mastercard e Visa) à matrícula da viatura. O condutor, sem ter de sair da viatura, introduz o cartão bancário no terminal de pagamento e o sistema associa automaticamente a matrícula ao cartão bancário. As portagens incorridas serão diretamente debitadas na conta associada ao cartão. O condutor tem ainda a possibilidade de alterar matrículas ou cancelar a adesão, através de um *Call Center* ou no site da *Portugal Tolls*;
- Visitantes Via Verde - é um dispositivo eletrónico para aluguer temporário com consumo em função da utilização, que tem a vantagem de poder ser utilizado não só nas portagens eletrónicas como também nas autoestradas com portagens tradicionais. O dispositivo é associado a uma conta bancária, na qual são debitados automaticamente os custos de cada passagem. Tem um custo de 6 euros de aluguer na primeira semana e 1,50 euros por cada semana seguinte, ainda com 27,50 euros de caução (que serão reembolsados no momento de devolução do dispositivo, em boas condições). Este pode ser alugado em *Via verde*, lojas Via Verde ou áreas de serviço.

Os sistemas apresentados aparentam ser uma solução. No entanto a opinião dos clientes diverge. Para os visitantes que venham em turismo, os sistemas são razoavelmente satisfatórios, no entanto para visitantes de curta duração, presentes por motivos profissionais, o facto de a aquisição dos sistemas ter de ser maioritariamente presencial, causa bastantes transtornos.

2.7 Car Sharing

Anunciado como o futuro dos transportes públicos, o *car sharing* é um serviço de aluguer de carros já bastante popular em todo mundo, tendo maior incidência nos EUA. Em Portugal

o negócio está a desenvolver-se, o aluguer das viaturas é feito à hora com combustível e parquímetro incluídos e está disponível 24 horas por dia. O aluguer pode ser feito através de uma plataforma *web*, portanto acessível via computador, smartphone, etc., ou via *call center*. O cliente pode encontrar uma viatura ”*car sharing*” num dos parques distribuídos pela cidade. Uma das empresas que fornece esta solução em Portugal é a *Mob Carsharing* da Carris.



Figura 2.6: *Car Sharing* [2].

Este tipo de negócio é recente mas tem-se afirmado no mercado com 1,7 milhões de utilizadores em cerca de 27 países [17]. A ZipCar, a maior empresa do ramo que conta com cerca de 80 mil clientes tem cerca de 9 mil viaturas e cresceu 44% no ano de 2012. Esta destaca-se pelo facto de ter sido uma das primeiras a chegar ao mercado e pelo tipo de vantagens que oferece, como a escolha do tipo de viatura que o cliente deseja (desde o mais económico ao mais luxuoso) e o melhor que existe em tecnologia para localizar viaturas livres, por exemplo via aplicação móvel.



Figura 2.7: Aplicação móvel *GetAround* - consiste numa espécie de rede social onde os seus utilizadores podem partilhar e alugar os seus veículos [3].

No início do ano, segundo o *New York Times*, a ZipCar foi adquirida pelo gigante do *rent car* tradicional, Avis Budget Group por 491 milhões de dólares americanos [17]. No entanto esta tem fortes adversários como a Car2Go que garante vantagens como estacionamento grátis no centro das cidades americanas e disponibilização de uma viatura pretendida perto do cliente no máximo de 15 minutos.

O *car sharing* é uma alternativa que começa a ganhar adeptos, Figura 2.7, visto que permite ao utilizador a redução de despesas pessoais e complementaridade com o transporte público, e mais valias para as redes de transportes, pois proporciona o descongestionamento de trânsito, a redução da ocupação do espaço urbano e consequentemente a redução de poluição.

Este tipo de soluções poderão sair prejudicadas com a existência de sistemas de portagens, como o ORT, onde o objectivo é facilitar a circulação e reduzir o desperdício de tempo, e acaba resultar no efeito contrário, envolvendo os utilizadores em processos de pagamento manuais lentos, ou pagamentos automáticos que poderão pouco intuitivos.

2.7.1 *Car Sharing vs* Sistemas de portagem ORT

O processo de aluguer de um automóvel pode facilmente tornar-se complicado. O cliente deve-se informar junto da companhia de aluguer quanto à forma de pagamento das portagens eletrónicas, uma vez que algumas empresas de aluguer de veículos, para comodidade dos seus clientes, instalam dispositivos eletrónicos para pagamento de portagens nas suas viaturas, repercutindo esses custos nos valores a cobrar.

Em Portugal, se os veículos não tiverem dispositivo instalado são normalmente os clientes que terão de efetuar o pós-pagamento num balcão autorizado, só disponível após 48 horas e durante um período de 5 dias úteis com a adição de 25 cêntimos por pósito (até um máximo de 2 euros), justificada em custos administrativos.

No caso de não efetuar pagamento, terminado o prazo legal, o dono da viatura incorre numa situação de infração e ser-lhe-ão debitadas coimas acrescidas de custos administrativos. Portanto neste caso, a companhia de aluguer receberá uma carta com a infração incorrida e a respectiva coima.

Nas duas possibilidades, com e sem dispositivo eletrónico, não existe forma de identificar e calcular de imediato quanto o cliente, neste caso de *rent car* ou *car sharing*, tem de pagar quando entrega o veículo, o que leva as companhias de aluguer a não colocar dispositivos eletrónicos nos seus veículos. Ou seja incumbem a responsabilidade de pagamento em balcão autorizado passado 48 horas aos seus clientes. Por indisponibilidade e falta de informação, os clientes poderão não pagar as portagens o que acarreta problemas para a companhia de aluguer automóvel.

Concluindo, o sistema atual exige uma forte gestão na seleção e atribuição do veículo a determinado cliente. A despesa apresentada por determinado veículo a determinada hora, terá de ser atribuída corretamente ao respectivo cliente. Atualmente, perante os sistemas envolvidos, este facto envolve custos operacionais e em recursos humanos para as companhias de aluguer. Mais, no modelo *car sharing*, a filosofia de utilização é totalmente deturpada e poderá resultar no decréscimo de clientes.

Do ponto de vista da rede de transportes é necessário criar sistemas que promovam a livre circulação de tráfego, proporcionem o descongestionamento e diminuam os níveis de poluição. Portanto é necessário evoluir de estratégia e melhorar os sistemas ORT atuais.

2.8 Futuros modelos e sistemas: o *Congestion Pricing*

Existem ainda outros pontos passíveis de melhoria nos sistemas ORT, tal como a criação de um interface para o utilizador com informações relacionadas com o preço das portagens, abrindo assim a possibilidade para a aplicação de um custo de portagem dinâmico, onde os factores de atribuição de preço de portagem podem não ser só a classe do veículo como também o nível de congestionamento da via por exemplo.

Esta visão intitula-se por *Congestion Pricing* e foi apresentada pelo Nobel da economia William Vickrey em 1952 [18]. Inicialmente, este propôs um modelo para o metro de Nova Iorque onde os preços de utilização aumentavam nas horas de ponta, quando a procura fosse elevada, baixando nas horas em que não havia tantos utilizadores. Embora aceite, as entidades responsáveis não conseguiram aplicar este modelo devido a limitações tecnológicas. Mais tarde este modelo foi proposto para a rede rodoviária. Em suma o objectivo é forçar o equilíbrio entre os períodos de elevada e baixa procura, de forma a que os utilizadores tenham a possibilidade de utilizar horários com portagens mais baratas, diminuindo de certa forma

o volume de congestionamento e consequentemente os níveis de poluição e de desperdício de energia, aumentando ainda o nível de utilização das infraestruturas, Figura 2.8.

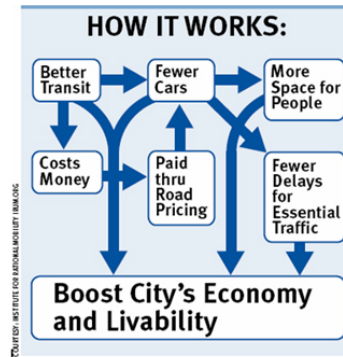


Figura 2.8: Modelo do *Congestion Pricing*.

Existem atualmente mais projetos relacionados com o *Congestion Pricing*, que tentam ainda ir mais longe, propondo de um modo vanguardista incorporar outro tipo de factores, como o nível de poluição gasosa e sonora ou o número de ocupantes da viatura, como por exemplo a proposta de um modelo de preços dinâmico no Reino Unido em *GNSS-based Dynamic Road User Charging System* [19].

Idealmente, a estratégia que se pretende tomar baseia-se num pagamento por um sistema de gestão integrado. O conceito é inovador, ambicioso e implementável num futuro próximo, onde a rede de transportes é gerida através de informação disponibilizada aos utilizadores e vice-versa. O objectivo é proporcionar ao utilizador um guia de viagem de tempo real com diversas funcionalidades, tais como:

- cálculo de custos de determinada rota e o seu equivalente em transportes públicos;
- registo de portagens incorridas;
- possibilidade de acesso a plataforma virtual de pagamento de portagens;
- gestor de avisos de acidentes/incidentes;
- cálculo de tempo estimado de rota baseado no tráfego atual;
- gestor de recuperação de veículo roubado;
- aplicações interativas, por exemplo localização de viatura num parque de estacionamento;
- entre outras.

Do ponto de vista da rede rodoviária o objectivo principal é incentivar os utilizadores a escolher transportes mais amigos do ambiente e mais seguros para uma determinada rota. Portanto esta estratégia lida diretamente com o utilizador como entidade pessoal, ou seja, enquanto que nos outros modelos e estratégias o utilizador poderia ser equiparado a um veículo, nesta estratégia existe interesse em comunicar diretamente com o utilizador condutor, podendo vários destes utilizar o mesmo veículo. Esta mudança de princípios, faz com que a

tecnologia a utilizar vá de encontra aos verdadeiros utilizadores. A tecnologia necessária a este modelo envolve sistemas complexos com comunicações estrada-veículo-utilizador (condutor), ou seja, a inclusão de mais um interface entre o utilizador/condutor e o sistema de pagamento que possa fornecer informação em tempo real, por exemplo um *smartphone*.

2.9 Soluções no mercado

A introdução de soluções mais flexíveis para o utilizador, através de soluções baseadas em *smartphone* tem sido debatida entre os produtores de tecnologia de pagamento de portagens [20]. As opiniões dividem-se, mas os mais crentes já estão a apostar em soluções integradas nos *smartphones* recorrendo a tecnologia Near Field Communication (NFC). Atualmente já existem serviços de pagamento por contacto, com *smartphone*, no Reino Unido, onde se aposta bastante na criação de uma entidade independente e agregadora de serviços de pagamento por contacto NFC, intitulada TSM (*Trusted Services Manager*), conceptualmente e funcionalmente semelhante a uma entidade do nível EETS dos pagamentos de portagem, mas que poderá ser fornecida por uma operadora que já tem serviços de apoio e suporte 24/7. Para muitos, um modelo que utilize única e exclusivamente o *smartphone* como meio de coleta de portagens, representa o futuro dos sistemas de pagamento, onde se poupa um dispositivo como o OBU e dá-se relevância aos aparelhos eletrónicos de consumo, com que o utilizador já se encontra familiarizado. Por outro lado a flexibilidade dos sistemas de comunicações móveis para operar em diferentes regiões ou países, através das funcionalidades de *roaming* leva a que esta solução comece a ganhar força. No entanto, existe a questão da segurança, que muitos consideram que é algo já ultrapassado e um não assunto, argumentando com a importância dos cartões Subscriber Identity Module (SIM) para a identificação de um utilizador e um telemóvel de forma unívoca numa rede, com total robustez, podendo disponibilizar toda a informação necessária ao fornecedor de serviço, neste caso companhia de portagens.

As opiniões mais optimistas antevêm que dentro de 5 anos, já existam soluções deste tipo no mercado, provavelmente integradas em sistemas telemáticos de novos veículos, que suportem comunicação NFC e utilizem o *smartphone* como um interface do utilizador como coordenador do veículo. Contudo, os mais reticentes afirmam que demorará 10 anos ou talvez mais (apontam para 2025), para que estes sistemas comecem a circular pelo mercado dos pagamentos de portagens, corroborando que os sistemas atualmente em operação não fornecem qualquer tipo de suporte para este tipo de tecnologias e que a integração levará alguns anos a ser desenvolvida. No entanto todos concordam que o mercado de produção e distribuição das *tags* tradicionais tem o fim à vista, com o crescimento exponencial do mercado dos *smartphones* e consequentemente da utilização da tecnologia NFC para pagamentos.

No início do presente ano de 2013, foi lançada uma solução nos Estados Unidos de América, que permite a consulta e o respetivo pagamento de portagens incorridas, através do *smartphone*, denominada GeoToll [21]. Os seus criadores anunciam-na como a primeira solução com um *smartphone* atuando como uma *tag* e portanto constituindo um *smartphone*-OBU. Este sistema utiliza *tags* 6C, pertencentes ao *standard* ISO/IEC 18000, operando na gama entre os 860 MHz e 960 MHz, sendo conhecidas por *EPCglobal Class 1 Gen 2* e utiliza ainda um módulo NFC, ambos integrados numa capa, ou luva, que envolve um *smartphone* (Android), tipo uma capa de proteção de borracha, e comunica com uma aplicação móvel. Portanto esta capa funciona como uma ponte de comunicação entre o pórtyco, que comunica com a *tag* 6C, e o *smartphone* pelo NFC. O sistema foi desenvolvido para utilizadores que

consultam as suas contas por *smartphone*, viajam muito e partilham o seu meio de transporte, como o *car sharing* ou o *car pooling*, a chamada "boleia". Já foram desenvolvidos dois protótipos e diversos testes, sendo apontadas algumas debilidades técnicas como a fraca autonomia que a aplicação origina no *smartphone*, falha solucionada com o recurso às funcionalidades do dispositivo e hardware que este possui, como o GPS que "acorda" o *smartphone* e coloca a aplicação em *foreground* quando se aproxima de um pórtico, desligando-o quando se afasta. Portanto o facto de se desenvolver soluções em plataformas providas de tecnologia, como o *smartphone*, representa uma vantagem no sentido de se flexibilizar e criar soluções para futuros problemas ou falhas. Sendo a interoperabilidade dos sistemas considerada uma falha, esta solução pode ser desenvolvida para se tornar interoperável, muito também pelas características e capacidades das tecnologias utilizadas.

A solução apresentada é considerada por muitos vanguardista. Atualmente os OBUs mais utilizados são robustos a falhas e simples de utilizar, no entanto são exclusivos de cada tecnologia e portanto não são capazes de interoperar com sistemas baseados em tecnologias distintas (DSRC vs glsGNSS-GSM).

2.9.1 OBUs baseado em GNSS e comunicações móveis GSM – GPRS com módulos DSRC

A empresa Alemã Continental AG apresenta várias propostas de OBUs, tal como ilustra a seguinte Figura 2.9:

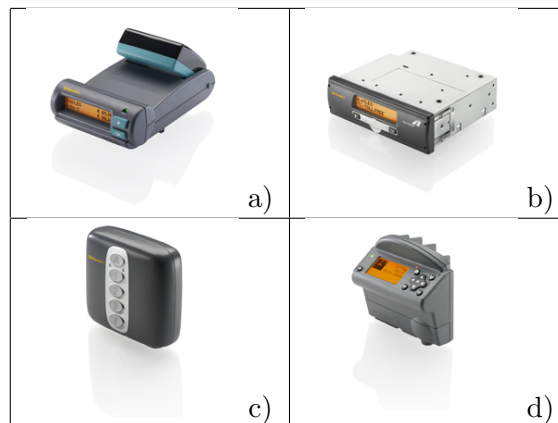


Figura 2.9: **a)** OBU 1372 – *Dashboard Solution*; **b)** OBU 1373 – *DIN Slot Solution*; **c)** OBU 1374 – *Windscreen Solution*; **d)** OBU 1376 – *The Swiss LSV* OBU [4].

Estes OBUs possuem: localizador GPS, sensor giroscópio, módulo de comunicações móveis GSM/GPRS, antenas para DSRC e interface com o utilizador. O DSRC utilizado serve para *enforcement* e aumento da precisão na aquisição da posição em alguns troços de estrada, como por exemplo cruzamentos ou viadutos. A Continental AG aplica os seus sistemas em diferentes redes de transportes um pouco por todo mundo, sendo os mais relevantes a Alemanha, Suíça, Índia e Austrália. E está a implementar um sistema de pagamento por zona ou área em Londres e tem como principal cliente os veículos pesados.

A Fela Management AG apresenta o TRIPON, soluções à medida do cliente, maioritariamente baseadas em GNSS/GPRS mas que também poderão ser implementadas em sistemas DSRC. Estão presentes área de transporte de veículos pesados na Alemanha, Suíça, Áustria,

Itália, França, Espanha e Portugal.

Um dos produtos é o TRIPON-CH, ilustrado Figura 2.10.



Figura 2.10: TriponCH.

Outra solução é o TRIPON-EU, Figura 2.11. Ambos utilizam tecnologias semelhantes aos produtos da Continental AG, no entanto destacam-se pelo interface para o utilizador, a possibilidade de atuarem como tacógrafo e a utilização do *smart car* para autenticação e registo de informações de viagem de determinado utilizador/conductor [22–24].



Figura 2.11: TRIPONEU.

Existe ainda uma solução mais simples, o TRIPON-SM [25], Figura 2.12, instalável na viatura pelo utilizador e que possui as funcionalidades mínimas de utilização. Tem ainda um interface bastante simplificado, com um botão e 4 LEDs, para que o utilizador possa declarar a classe do veículo que conduz e possa receber mensagem de erros.

No mercado a empresa Austríaca Kapsch apresenta um sistema híbrido, Figura 2.13, sendo este o que está mais próximo de um sistema interoperável.



Figura 2.12: TRIPONSM.



Figura 2.13: hybridOBU

O OBU TS3209/00A [26], apresenta as tecnologias necessárias para o funcionamento nos dois sistemas de portagens mais comuns: localizador GPS, módulo de comunicações móveis GSM/GPRS e antenas para DSRC. O interface com o utilizador é semelhante ao TRIPON-SM da Fela Management.

Os OBUs apresentados são um bom exemplo de que já existem dispositivos que possuem as duas tecnologias integradas. Estes são comparados na tabela presente no Anexo B. Conclui-se que as soluções são bastante semelhantes a nível tecnológico. De notar que praticamente todos necessitam de alimentação pelo isqueiro do automóvel e que uns são mais portáteis e instaláveis pelo utilizador do que outros. Contudo este tipo de soluções foram desenvolvidas para veículos pesados. Em suma, estes sistemas representam um bom início para o desenvolvimento de OBUs interoperáveis na Europa.

Capítulo 3

Charge Collector System

3.1 Introdução

O C2S constitui uma proposta de abordagem para um novo sistema de pagamento de portagens eletrónicas, em infraestruturas ORT, motivado principalmente pela ausência, nos sistemas atuais, de um nível de abstração entre o utilizador de um veículo e o pagador da portagem, e por outro lado pela inexistência de um interface gráfico nas tecnologias existentes que possibilite troca de informação entre o utilizador e o sistema.

O nível de abstração entre utilizador/pagador e o proprietário da viatura é uma questão importante para certos cenários. Atualmente o pagador de portagens é unicamente o proprietário do veículo. Em sistemas de *car sharing* esta associação unívoca pode levantar problemas, pois o veículo pode ser utilizado por diferentes utilizadores em curtos períodos de tempo (uma hora, por exemplo) e que terão de se deslocar a um balcão oficial para efetuar o pagamento, CTT por exemplo. Na tipologia do C2S propõe-se que, por defeito a entidade pagadora é o proprietário do veículo, no entanto existe a opção de um dos utilizadores pagar as portagens que efetuou, tal como a Figura 3.1 ilustra, de forma automática, via aplicação móvel.

Para incorporar esta funcionalidade surgiu a necessidade de integrar um interface gráfico capaz de informar o utilizador das portagens já incorridas pelo veículo e possibilitar a escolha de quais as portagens que o utilizador tenciona pagar, bem como o modo de pagamento que pretende.

Uma segunda motivação, não menos importante, para o desenho deste sistema é a necessidade de integração das tecnologias DSRC e GNSS-GSM. Desta forma o C2S está preparado para operar em ambas as tecnologias, base dos sistemas atuais mais utilizados, e poderá pertencer ao grupo de OBUs com estas características, apresentados no capítulo do Estado da Arte. Além da tecnologia que o OBU C2S possui, o facto do sistema se basear numa ligação à Internet para comunicar com os sistemas das companhias de portagem, torna a partilha de informação entre sistemas heterogéneos mais fácil e flexível. Com a integração de tecnologias e reajuste da estrutura dos sistemas de pagamento, a questão da interoperabilidade entre sistemas e a criação do EETS torna-se um desejo mais realista.

Em traços gerais, o sistema C2S oferece ao utilizador um interface gráfico acessível pelo seu *smartphone* via aplicação móvel, com opções de consulta e pagamento de portagens, e por outro lado ambiciona proporcionar aos condutores não proprietários do veículo (veículo alugado, por exemplo) a possibilidade de pagamento de portagens no final da sua viagem.

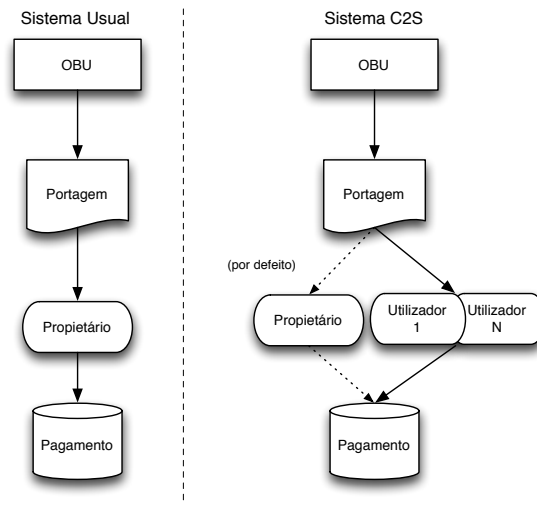


Figura 3.1: Ação de pagamento multi utilizador/pagador com Proprietário definido por defeito.

Do ponto de vista conceptual, o sistema C2S representa na arquitetura global de sistemas de pagamento a unidade responsável pelo registo das portagens incorridas pelo veículo. Em complemento é necessário utilizar um segundo dispositivo provido de um módulo de comunicações, que permita a ligação entre o sistema instalado na viatura e o eventual *back-office* do sistema global de pagamento.

Para o segundo dispositivo poderá ser utilizado um *smartphone*, pois é tecnologicamente capaz e é amplamente utilizado. O sistema C2S poderá completar-se assim, com um aparelho eletrónico pessoal, com o qual o utilizador está familiarizado e poderá tirar partido de diversas funcionalidades.

O tipo de perfil indicado para esta solução é o utilizador de *smartphone*, que utiliza o mesmo para pesquisar e alugar veículos num regime de curta duração (poucas horas), ou seja num formato *car sharing*. Com a difusão dos *smartphones*, a crescente oferta de soluções de *car sharing* como transporte público, bem como a aplicação de sucessivas medidas contra a utilização de veículos particulares nas grandes metrópoles, leva a que o utilizador alvo desta solução comece a considerar-se significativo.

Do ponto de vista tecnológico o sistema C2S é constituído por um OBU com função de um *data logger* capaz de: detectar pórticos ou zonas de portagens, portanto operável em infraestruturas DSRC e GNSS-GSM; e transferir os registos efetuados, ditos *logs*, para um outro dispositivo, utilizando um sistema de comunicação sem fios de curta distância (NFC, *Bluetooth* ou WiFi).

O sistema completa-se com um segundo dispositivo, móvel, provido da mesma tecnologia de comunicação de dados a curtas distâncias e com ligação à Internet, com um interface gráfico, capaz de recolher os *logs* do OBU (mesmo em *offline*) e possibilitar o pagamento das portagens *a posteriori*.

No presente capítulo irão discutir-se questões relacionadas com a arquitetura do sistema, isto é, o desenho e articulação dos blocos constituintes do C2S. Irão ainda expôr-se as tecnologias principais utilizadas e concluir-se com a demonstração da utilidade deste sistema com dois *user cases*.

3.2 Arquitetura dos Sistemas de Pagamento

A arquitetura e a estrutura dos sistemas de pagamento de portagens dependem do tipo de estratégia de portagem aplicada, tema discutido no capítulo do Estado da Arte. Entenda-se por arquitetura o tipo de estrutura, isto é, distribuição dos diferentes componentes e módulos do sistema, e todo o fluxo de operação.

O tipo de tecnologia e a dependência da ação humana (operadores) são fatores determinantes para a complexidade da arquitetura e da estrutura a implementar. Os sistemas de estrada, baseados em DSRC, totalmente automáticos (por exemplo sistemas Multi-Lane Free-Flow (MLFF)) têm de possuir mecanismos de fiscalização e auditoria igualmente automáticos, além disso estas procuram um nível de eficácia semelhante ao sistema manual, isto torna a arquitetura mais complexa e consequentemente o investimento inicial na estrutura tecnológica mais avultado.

Atualmente, a estrutura dos sistemas MLFF mais usuais, dividem-se normalmente em três grandes blocos, como é o caso da estrutura do sistema MLFF que o grupo Ascendi tem em Portugal, constituído pelo: equipamento de estrada, centro de operações e departamento comercial. Este tem ainda um módulo de comunicações entre cada bloco. Um sistema com este tipo de arquitetura envolve entidades externas e diversos parceiros tecnológicos responsáveis por diferentes matérias, tal como ilustra a seguinte Figura 3.2.

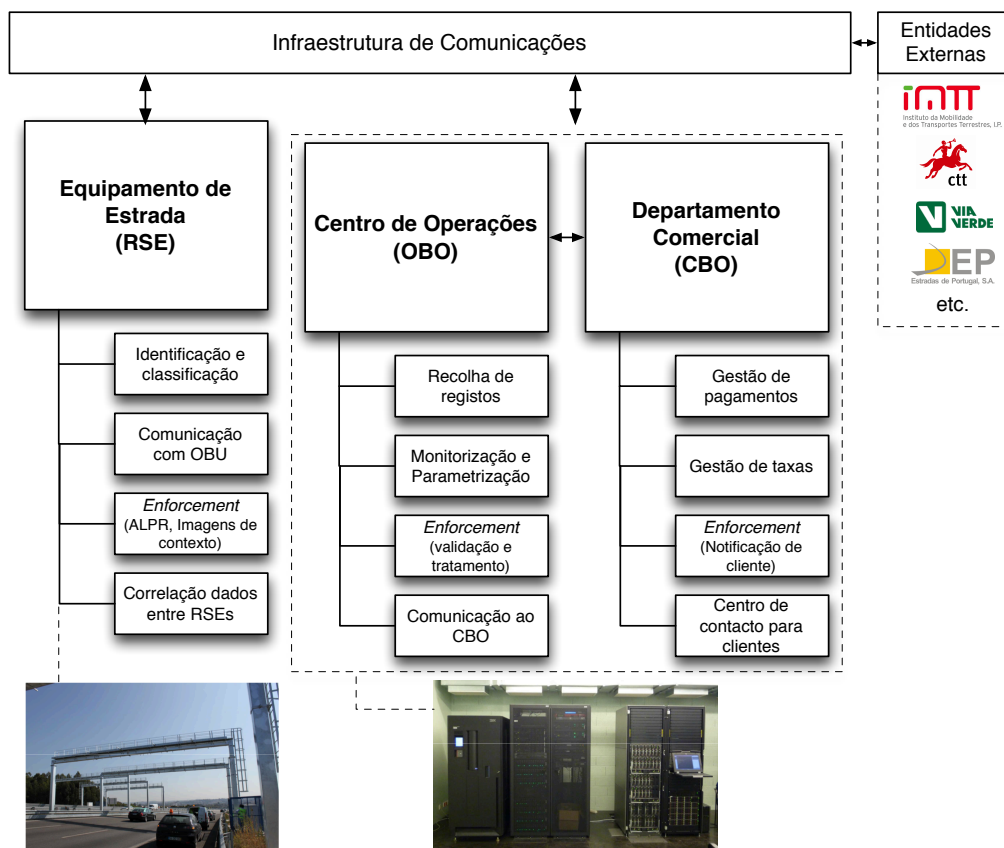


Figura 3.2: Arquitetura do sistema MLFF da Ascendi. Baseado e extraído de [5]

O equipamento de estrada ou RSE é o bloco base do sistema, funciona como *input* de toda a estrutura e é constituído por 4 subsistemas. O primeiro é um subsistema de identificação e classificação de veículos, responsável pela detecção da passagem do veículo e por aferir a classe do mesmo. O segundo é um subsistema de comunicação com os OBUs dos veículos. Existe ainda um subsistema de vídeo responsável pelo registo e fiscalização das passagens com captura de imagens e leitura da matrícula denominado ALPR. Por último existe um subsistema responsável pela comunicação entre RSEs, capaz de correlacionar dados e transmitir estes ao *back-office*.

O segundo bloco da estrutura é o centro de operações ou Operational Back-Office (OBO), que funciona como interface entre o RSE e o departamento comercial. Este é responsável pela recolha dos registos efetuados, pela distribuição das listas de taxas e listas de estados, bem como monitorização e parametrização dos RSEs. Por outro lado atua sobre a validação, tratamento de possíveis anomalias, discrepâncias ou violações, consolidação e recuperação das transações efetuadas, analisando os percursos realizados por determinado veículo. Finalmente, este estabelece um interface para comunicação com o departamento comercial, transferindo as transações efetuadas para a base de dados de cliente e oferecendo opções de configuração ou simulação de novas taxas de portagem a aplicar.

O terceiro e último bloco é o departamento comercial ou Comercial Back-Office (CBO), responsável pela gestão das taxas a aplicar, como receção dos pagamentos, pela gestão das contas de cliente e pela gestão de finanças. Por outro lado envolve toda a componente de cliente, como gestão de relacionamento com clientes, tratamento de possíveis reclamações e centro de contatos. É neste bloco que se aplica o último nível do processo de fiscalização e auditoria, onde se tem acesso aos dados dos proprietários dos veículos e se processa todas as notificação de clientes, em cooperação com entidades externas, como o Instituto da Mobilidade e dos Transportes Terrestres (IMTT), os CTT, Correios de Portugal, a Via Verde, as Estradas de Portugal e outras.

No caso de um sistema GNSS-GSM, o sistema é globalmente semelhante, sendo que as infraestruturas de estrada são utilizadas em muito menor número e apenas como instrumento de fiscalização ou de despistagem de possíveis erros. O sistema é fortemente dependente da tecnologia a bordo, para aquisição da posição do veículo, e do suporte de uma operadora para efetuar comunicações.

3.2.1 Arquitetura C2S

O sistema C2S é maioritariamente externo ao sistema apresentado. Este representa um novo tipo de registo de veículos que atua como um *input* ao sistema global. Portanto os clientes submetem/declaram que incorreram em portagens e o sistema coordena o processo para efetuar o pagamento, junto do departamento comercial. Para isso é necessário um interface extra que disponibilize Application Programming Interface (API), para estabelecer a comunicação entre a aplicação móvel do cliente e o *back-office*. Uma visão sobre o sistema global está representada na Figura 3.3.

Assim, a estrutura do C2S é composta por um primeiro bloco, denominado OBE, por um terceiro bloco, denominado vulgarmente *back-office*, capaz de receber registos e pagamentos de portagens, e por um segundo bloco de comunicação, assente numa aplicação móvel que estabelece “a ponte” entre as duas anteriores. O OBE é constituído por um computador embebido, um receptor GPS, um detector de pórtico DSRC e um interface de comunicação sem fios, e é responsável pelo registo das portagens incorridas, ou seja, possui uma base de

dados (com as coordenadas das zona de portagem atribuídas a um número identificador, ou ID), processamento e armazenamento local, para inferir se as posições por onde o veículo viaja são locais de portagem. No fundo, este bloco opera como um *data logger*, sensível a zonas de portagem e possui um interface de comunicações sem fios para enviar os *logs* para outro dispositivo.

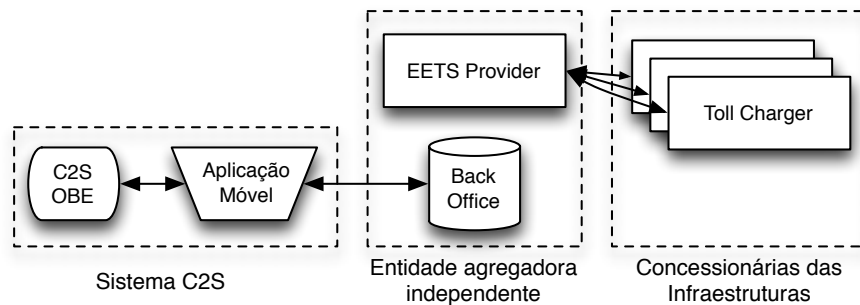


Figura 3.3: Estrutura global de operação do C2S e das restantes entidades.

O segundo bloco é o dispositivo móvel onde existe uma aplicação em execução. Este é responsável pela recolha do registo de portagens, por estabelecer um interface com o utilizador, onde são apresentadas as portagens incorridas com os respetivos preços e pela comunicação das portagens selecionadas ao sistema de pagamento. Este deve ainda apresentar opções de seleção do tipo de pagamento eletrónico que o utilizador pretende utilizar (Paypal, cartão de crédito, por exemplo) e de pedido de fatura eletrónica à companhia de portagens. A aplicação móvel possui uma base de dados existente no *back-office* do sistema de pagamento, onde atribui ao número identificador (correspondente ao do OBE) diversa informação relativa à portagem como o nome, infraestrutura, concessão, preço, etc.

Portanto as duas base de dados não possuem a mesma informação, a primeira (instalada no OBE) associa coordenadas a um ID e a segunda (armazenada na aplicação móvel) associa um ID à descrição de uma portagem (com nome, preço, etc.). A existência de duas base de dados locais, justifica-se em primeiro lugar pelo facto de o OBE ser um sistema *standalone* que estabelece comunicação com o *back-office* de forma ocasional e assíncrona, mediante iniciativa do utilizador (quando este lê o OBE com a aplicação móvel) e em segundo lugar pelo facto de que, ao se associar coordenadas a um código identificador (ao ID), consegue-se diminuir a quantidade de informação a transferir e aumentar a imunidade a possíveis erros de transmissão. O facto destes dois blocos (OBE e a aplicação móvel) não terem uma comunicação síncrona, ou seja não estarem continuamente ligados entre si e entre eles e o *back-office*, também constitui um forte motivo para que possuam armazenamento e base de dados locais, de forma a garantirem persistência e robustez nos dados. A comunicação entre os dois blocos estabelece-se da forma ilustrada na Figura 3.4.

O OBE envia os registos de portagens que estão no ficheiro de *logs* para a aplicação móvel, e a aplicação móvel responde com um *acknowledge* pelo envio das portagens pagas, guardadas internamente no *smartphone*. Depois da transferência estabelecida, o OBE lança um processo, responsável por registar a portagem paga num buffer circular e apagar o registo de portagem no ficheiro de *logs*. No caso de o OBE não receber o *acknowledge* os *logs* persistem. Ou seja, só depois do segundo contacto, da aplicação móvel com o OBE, é que o registo de uma determinada portagem é apagado e assim se completa o ciclo.

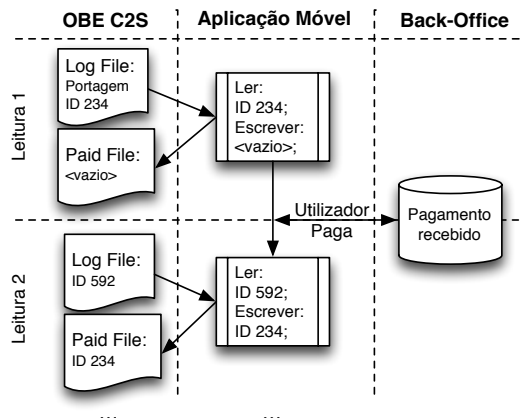


Figura 3.4: Protocolo de comunicação entre OBE, a aplicação móvel e o *back-office*.

O facto de se utilizar duas bases de dados suscita alguns problemas. O primeiro e mais importante, surge na necessidade de manter as bases de dados atualizadas, ou seja, quando e como desencadear processos de atualização dos dados, principalmente na base de dados mais “distante” do *back-office*, a do OBE. Para isso propõe-se que a atualização seja desencadeada pela transmissão de uma mensagem de *acknowledge* de pagamento concatenada com as portagens a atualizar, tal como ilustra a Figura 3.5.

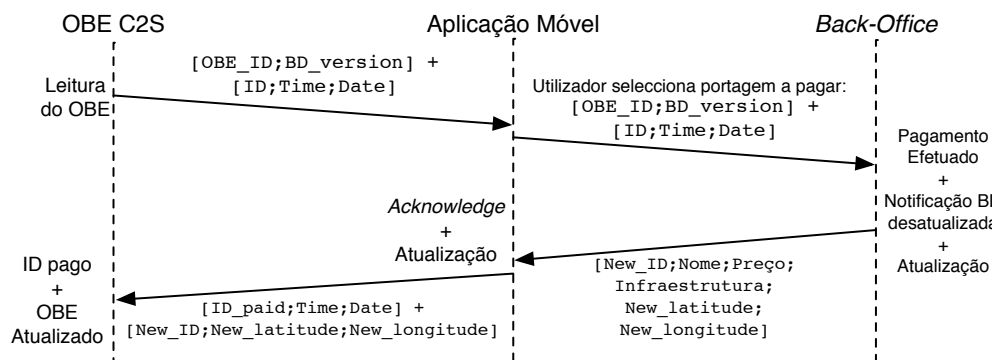


Figura 3.5: Proposta de modelo para atualização das base de dados.

O OBE ao receber as mensagens e identificar que não se trata de registos de portagens pagas: [ID;time;date] mas novas posições de portagens: [ID;latitude;longitude], escreve as mesmas na base de dados. Ou seja, a aplicação móvel terá de receber informação especial do *back-office* com as coordenadas e ID das portagens que são para atualizar o OBE. A atualização da aplicação móvel pode ser realizada quando esta se liga ao *back-office*. Durante a declaração de pagamento o *back-office* questiona qual a versão da base de dados que a aplicação possui localmente, no caso de estar desatualizada, o utilizador é informado e obrigado a atualizar a base de dados local.

Outra questão importante, reside no facto de as infraestruturas portajadas poderem pertencer a diferentes concessionárias. Perante este facto e na eventualidade de as concessões serem tecnologicamente independentes, surge a necessidade de criar um terceiro bloco no sis-

tema serviço de *back-office*, capaz de comunicar com a aplicação móvel. Este deve ter uma base de dados global, onde integra a informação de todas as concessões. Pois bem, este serviço deve ser prestado por uma entidade independente que agregue toda a informação de todos os pórticos, a nível nacional ou Europeu por exemplo, e realize transações com as respetivas concessionárias de estrada. Este tipo de estrutura equipara-se assim à estrutura do EETS, onde existe a definição de uma entidade de cobrança a nível do *EETS Provider*, ou seja uma entidade independente que comunica com as respetivas entidades concessionárias, ou *Toll Chargers*, das infraestruturas portajadas, ou *EETS Domains*. Segundo a última atualização do registo nacional do EETS em Portugal, ainda não existe nenhum fornecedor de serviço com características de *EETS Provider*, mas a nível nacional existem duas entidades a nível nacional capazes de realizar tal tarefa, a Via Verde,SA e os CTT – Correios de Portugal,SA.

O facto de o terceiro bloco do sistema, o *back-office*, estar instalado numa entidade independente das entidades concessionárias, poderá fornecer serviços capazes de diminuir o peso computacional da aplicação móvel, como por exemplo: particionar a base de dados geral, isto é, fornecer "uma janela" da base dados indicada para determinado contexto, pois se um utilizador se encontra em Portugal pode libertar memória local, ocupada com informação das infraestruturas do norte e centro da Europa, por exemplo. A aplicação móvel deve assim ser controlada por um *back-office* exclusivo para o C2S e sob gestão de um *EETS Provider*, ficando este com a responsabilidade de agregar e transacionar toda a informação necessária com os *Toll Chargers*, tal como fará quando o EETS estiver em operação.

Sendo o C2S um sistema de recolha de informação para pagamento, portanto sendo este obrigatório e não voluntário, é importante criar mecanismos, no desenho da estrutura, que garantam fiscalização e auditoria dos pagamentos e registos, os chamados mecanismos de *enforcements*. O C2S baseia-se nos sistemas de *enforcement* já em operação, das infraestruturas DSRC e GNSS-GSM, como o sistema ALPR, sendo abordado na secção do Modo de Utilização do presente capítulo. É importante salientar, que estando este projeto numa fase de prova de conceito, são propostos mecanismos de *enforcement* mínimos, passíveis de melhoria.

Portanto, em tom de síntese, a arquitetura do sistema C2S assenta em três blocos principais: o OBE, a aplicação móvel e o *back-office*, todos com funções e características diferentes, e com três base de dados distintas. Para o sistema pretendido, considera-se que este tipo de estrutura representa uma boa base para a implementação dos blocos fundamentais.

3.3 Tecnologia

No geral, um sistema de pagamento de portagens é composto por diversos componentes tecnológicos. Os dois mais importantes do sistema são os responsáveis pelo reconhecimento de veículo e pela identificação de conta de cliente. O reconhecimento dos veículos pode ser executado de várias formas, utilizando diversas tecnologias, desde pórticos de estrada equipados com câmaras, sensores e sistemas de comunicação veicular, a infraestruturas globais baseadas em tecnologias GNSS e comunicações GSM. O reconhecimento pode-se estender à classificação da classe do veículo em causa, envolvendo assim sensores adicionais. A identificação de conta é feita consultando uma base de dados relacional com o ID do veículo identificado.

Nesta fase de projeto, o sistema C2S foca-se essencialmente no registo do veículo em zonas de portagem e na criação de condições para a concepção do módulo de comunicação com o sistema de pagamento. Para isso, o registo de veículo do C2S assenta o seu funcionamento num sistema de navegação por satélite (GPS por exemplo) e um OBE provido de tecnologia

para processar e interpretar a informação recolhida.

O OBE C2S necessita de um sistema computacional com capacidade de executar diversos processos ao mesmo tempo, para processamento das coordenadas recebidas via GPS, de modo a detectar se estas pertencem a zonas de portagem, e por outro lado para efetuar *logs* e comunicar com outro dispositivo. Além do receptor de GPS, outro periférico importante é o módulo de comunicação sem fios, que se poderá basear em qualquer das tecnologias de conectividade mais utilizadas, Universal Serial Bus (USB), WiFi ou Bluetooth, ou até uma menos utilizada mas em expansão como o NFC. Por último, para mecanismo de redundância, segurança e futura implementação de mecanismos de *enforcement*, o sistema poderá ter um detetor de pórticos DSRC, conectado com um General Purpose Input/Output (GPIO). Para conclusão do projeto OBE é ainda importante ter em conta LEDs de sinalização de modos de funcionamento, um *buzzer* para sinalizar passagem em zona de portagem, um botão de reset do sistema e a forma de alimentar eletricamente o sistema.

Quanto ao segundo dispositivo do sistema C2S, é importante que este esteja provido de um módulo de comunicações sem fios a curtas distâncias, seja capaz de receber informação do OBE, disponibilizar esta num interface para o utilizador e comunicar a informação selecionada ao *back-office* via Internet. As características e tarefas desejadas para este dispositivo podem ser executadas por um *smartphone* provido de um sistema operativo móvel (Android, por exemplo). Mais, desta forma as funcionalidades deste módulo são escaláveis e facilmente expansíveis a novos cenários de utilização, podendo criar-se novas soluções, como seleção da conta de pagamento ou disponibilização de informação sobre congestionamento de trânsito ou preços de combustíveis.

3.3.1 Sistemas de navegação por satélite

Os sistemas de navegação por satélite são utilizados num vasto leque de aplicações. Estes começaram a ser testados nos sistemas de pagamento de portagens na Alemanha em 1994, e em 1998 a União Europeia propôs a utilização deste tipo de sistemas nas novas infraestruturas criadas, devido às suas vantagens relativas à não utilização de infraestruturas de estrada, monetariamente pesadas, e à facilidade de instalação, gestão e atualização. O interesse pelo sistema aumentou e em 2005 a Alemanha implementou o primeiro sistema de pagamento de portagem por satélite, baseado na distância percorrida, para veículos de transporte de mercadorias pesadas (maiores que 12 toneladas).

Ainda hoje a Europa demonstra empenho no desenvolvimento deste tipo de sistemas, no que concerne às portagens pagas eletronicamente, devido não só às vantagens económicas e funcionais já referidas, mas também pelo facto do sistema Galileo estar já em fase de teste, sendo esta aplicação uma forma de rentabilizar o forte investimento nesta infraestrutura europeia, equipada com tecnologia necessária para responder ao desafio.

Atualmente, os sistemas de portagem baseados em navegação por satélite utilizam o conhecido sistema americano GPS, na grande maioria dos casos. Este funciona através de uma rede global de satélites, os quais transmitem sinais rádio para os utilizadores, e estes com os respetivos receptores conseguem obter a posição no tempo e no espaço. Com um receptor GPS é possível calcular a distância a que nos encontramos de um satélite, multiplicando o tempo medido de propagação e a velocidade da propagação do sinal rádio (velocidade da luz). Conjugando o sinal de 3 satélites, através da técnica de triangulação, consegue-se inferir a posição do receptor. Embora pareça um processo simples, o sistema GPS exige uma elevada precisão temporal e sincronismo entre satélites, o que torna o seu funcionamento complexo.

Para casos de necessidade de precisão posicional mais elevada, na ordem dos centímetros, é frequentemente utilizado GPS diferencial.

O sistema GPS está dividido em três segmentos: o segmento espacial, constituído pela constelação de 24 satélites, colocados a 20,2 mil km de altitude e rodando duas vezes por dia em torno da Terra. A constelação, garante que pelo menos 4 satélites estão “em linha de vista” de qualquer ponto da Terra [27]; o segmento de controlo, constituído por uma central de controlo principal, 12 antenas de controlo e 16 centros de monitorização, que comunicam diretamente com o segmento espacial, garantindo a eficiência e precisão do sistema; e o segmento do utilizador, onde são desenvolvidas aplicações para uma panóplia de áreas, desde a aviação à indústria automóvel.

Existem ainda outros sistemas de navegação por satélite, como o Globalnaya Navigatsionnaya Sputnikovaya Sistema (GLONASS) pertencente à Rússia, o European Geostationary Navigation Overlay Service (EGNOS) junção do GPS e GLONASS, e o Galileo. O Galileo é o sistema promissor da Europa para o mundo, desenvolvido para fins comerciais e aberto à cooperação internacional. Para a indústria automóvel, este sistema (independente de outros sistemas) irá fornecer serviços exclusivos, como o *Galileo Open Service*, o *Commercial Service* ou o *Support to Search and Rescue Service*, que prometem ter performance e disponibilidade maior que as atuais, operando mesmo cenários indoor.

3.3.2 Computador Central

Como módulo principal, integrador dos periféricos do OBE, é desejável utilizar um microcontrolador com hardware que suporte multiprocessamento e execute software que permita *multitasking*, dadas as tarefas que são necessárias desempenhar por este módulo. Para aplicações na área da indústria automóvel são muitas vezes aplicados sistemas computacionais dedicados, chamados sistemas embebidos, para determinada tarefa, pouco complexos e leves, para cumprir janelas temporais apertadas. No sistema que se pretende desenvolver, os constrangimentos temporais exigidos não são rigorosos, é sim necessário uma plataforma flexível, atualizável e passível de integração em sistemas já existentes (do Intelligent Transportation Systems (ITS), por exemplo).

Portanto, para o desenvolvimento do sistema decidiu-se utilizar um computador de dimensões reduzidas, do tamanho de um cartão de crédito e com baixo custo, com todo o *hardware* necessário disponível numa só placa e com software versátil. Depois de desenvolvido um estudo das características do sistema que se pretendia criar e as soluções disponíveis no mercado, o Raspberry Pi (RPi) foi identificado como o mais indicado. Embora atualmente existam concorrentes cada vez mais fortes e o mercado dos *single board computers* esteja a ficar cada vez mais competitivo, o RPi desenvolvido nos laboratórios de computação da Universidade de Cambridge e distribuído pela Fundação RPi desde 2012, é uma opção indicada para o desenvolvimento de provas de conceito, dado o seu desígnio académico e didático, beneficiando de uma vasta comunidade de utilizadores com *know-how* em diversas áreas tecnológicas.

Tecnologicamente o RPi é baseada por um System on Chip (SoC) Broadcom BCM2835 que inclui um processador da família ARM11 de 700 MHz, uma memória RAM de 512 MB na última versão, não tem memória não-volátil, mas utiliza como alternativa um cartão SD para armazenamento de dados, e tem ainda hardware adicional que não se considera relevante neste momento para o projeto (GPU VideoCore por exemplo). Existem dois modelos no mercado, o A e o B, sendo o B uma expansão do A, em um *hub* USB (extensão USB) com duas portas USB e controlador Ethernet, sendo também mais caro (cerca de 10\$) e consumindo mais potência

(mais 1 W). Ambos são alimentados via conector MicroUSB a 5 Volts e contêm ainda portos para conexão de periféricos de baixo nível, através de GPIO) (com Inter-Integrated Circuit (I2C), Serial Peripheral Interface (SPI) e a possibilidade de ser eletricamente alimentados por estes portos. O sistema peca por não possuir Real Time Clock (RTC) embebido, tendo o sistema de utilizar um Network Time Protocol (NTP) ou ter de se ligar a um RTC externo via I2C por exemplo.

Uma das vantagens do RPi, advém do facto de este operar com sistemas operativos baseados em Linux e estes serem diversificados, contendo características importantes, como o peso computacional do sistema operativo e as ferramentas de programação suportáveis. Para se escolher qual o sistema operativo mais indicado, efetuou-se uma pesquisa, da qual resultaram as seguintes distribuições:

- **Arch ARM** é uma distribuição da Arch Linux, que prima pela simplicidade e pelo completo controlo para o utilizador final. Esta contém uma estrutura base leve que possibilita ao utilizador moldar o sistema conforme as suas necessidades. Por esta razão, a Arch Linux ARM para a RPi, por defeito, não vem com ambiente gráfico, no entanto é possível instalar um. Esta distribuição não é aconselhável para principiantes;
- **Raspbian** é uma distribuição baseada no Debian otimizada para o hardware da RPi. Este resulta de um projeto para criar um ARM hard float port (ARM hard float (ARMHF), possui melhor performance já que utiliza a unidade de floating point implementada no chip) da distribuição Debian Wheezy para a RPi. Neste momento o Raspbian é considerada a distribuição para RPi mais fácil de usar, mais estável e com melhor performance [28];
- **Debian** para o RPi é a referência do sistema de ficheiros de Cambridge, onde o dispositivo foi desenvolvido. Esta contém um ambiente gráfico (LXDE) e um browser (Midori), possui ferramentas de desenvolvimento para funcionalidades multimédia que o dispositivo possui. No entanto não utiliza ARMHF e poderá ser mais lenta [28]. O seu tempo de inicialização depende do tamanho e velocidade do cartão SD a que está associado. Fruto do desenvolvimento e sendo estes sistemas operativos em constante evolução, existem ainda as distribuições wheezy e lenny do debian que são as distribuições em desenvolvimento, ou seja instáveis, e as já obsoletas, respectivamente. A versão squeeze é a mais estável [29];
- **Fedora Remix** é uma distribuição composta por software do projeto Fedora ARM, mais alguns pacotes adicionais que foram desenvolvidos especificamente para a RPi. Esta possui ambiente gráfico, diversas linguagens de programação (Python, Perl, Ruby ou Bash) e ferramentas de sistema (Secure Shell (SSH)). Esta distribuição aparenta aproximar-se mais do computador normal, visto que contém aplicações como editores de texto, browser (Firefox) [30], etc.;
- **RISC OS** é o sistema operativo Britânico desenhado em Cambridge pelos mesmos engenheiros que criaram os processadores ARM [31]. Este sistema é compacto e eficiente, bastante utilizado para desenvolvimento de técnicas de programação. O RISC OS possui ambiente gráfico e algumas aplicações, cabe em 6 MB. Foi desenvolvido para um computador que tinha 8 MHz ARM2 e 512 KB de memória RAM. Portanto hoje em dia, este é muito rápido para as máquinas potentes que possuímos como a RPi. RISC

SO/Distribuição	Vantagens	Desvantagens
ARCH Linux	Simples, Rápido e Completo controlo do sistema; ARMHF.	Exige conhecimentos aprofundados da matéria e implementação de algumas funcionalidades não existentes por defeito.
RASPBIAN	Fácil de usar; Estável; Melhor Performance; ARMHF.	Mais direcionada para ambiente gráfico e aplicações multimédia.
DEBIAN	Bastante Documentação; Poderá ser rápido dependendo da versão e implementação.	Não possui ARMHF; Ambiente gráfico e aplicações multimédia.
FEDORA REMIX	Boa integração com diferentes linguagens de programação; Bastantes funcionalidades já implementadas.	Não possui ARMHF; Ambiente gráfico, comporta-se como um computador normal, podendo portanto ser lento
RISC OS	Pequeno; Compacto; Simples; Fácil de entender.	Não possui ARMHF; Difícil de alterar; Desenhado para um computador normal; Muito primitivo (poucas layers entre máquina e utilizador).

Tabela 3.1: Comparação entre Sistemas Operativos.

OS não é uma versão do Linux nem está relacionada com o Windows, é um sistema muito mais simples e básico. Recentemente foi apresentado o sistema RISC OS para RPi. Este aparenta ser um projeto embrionário que ainda não possui muito desenvolvimento e o facto de ser um sistema bastante primitivo com poucas camadas a separar o utilizador da máquina faz com que seja difícil alterar funcionalidades sem que o nível de complexidade aumente;

Embora existam mais de vinte distribuições a funcionar no RPi, as distribuições mais usadas, robustas e documentadas são as que foram expostas anteriormente. Para se proceder a uma melhor comparação de características, construiu-se a Tabela 3.1.

Dadas as características apresentadas na tabela, conclui-se que os dois sistemas mais indicados para o projeto em questão é o **Arch ARM** ou o **Raspbian**, onde o primeiro prima pela simplicidade e o segundo pela documentação e quantidade de projetos em que foi utilizado. A distribuição utilizada no sistema foi o Arch ARM.

3.3.3 *Smartphone*

O *smartphone* foi o dispositivo escolhido para efetuar a comunicação entre o OBE e o *back-office*, e proporcionar uma experiência dinâmica ao utilizador, em termos de opções de pagamento que queira tomar ou registos de portagens que queira consultar. Estes dispositivos da eletrónica de consumo, tornaram-se objetos pessoais cada vez mais comuns no dia-a-dia dos condutores.

As plataformas móveis têm evoluído a ritmo massivo. O termo *smartphone* substitui os conhecidos dispositivos Personal Digital Assistant (PDA), que disponibilizavam os primeiros sistemas de telemóvel combinados com funcionalidades de computador, graças às suas elevadas performances, quantidades de tecnologia que agregavam, como GPS, sensores de movimento, câmara, módulos de comunicação (Bluetooth, WiFi) e sistemas operativos móveis que incorporam. O mercado da computação móvel continua em expansão, fruto da elevada procura e da versatilidade de dispositivos existentes, como *smartphones*, tablets, câmaras fotográficas, ou dispositivos *wearable* (óculos, relógios, ou sapatilhas inteligentes). No início do ano 2013, a imprensa Americana, adianta que no seu mercado, os *smartphones* já ultrapassaram os chamados *feature phones*, e que atualmente o poder de computação de um *smartphone* banal é muito superior à tecnologia que ia a bordo do Apollo 11 e colocou o homem na lua [32].

Os líderes do segmento são o grupo Sul Coreano Samsung e a Norte Americana Apple Inc., segundo o New York Times [33], que utilizam dois sistemas operativos distintos, o Android baseado em Linux e desenvolvido pela Google Inc., e o iOS baseado em Unix desenvolvido pela Apple Inc., exclusivamente para dispositivos da marca, não sendo compatível com hardware de terceiros. Embora nas funcionalidades e características os dois sistemas sejam bastante semelhantes, isto é, permitem utilizar o conceito de aplicação que o utilizador compra, descarrega e instala com um simples toque, permitem *multitasking* ou *multitouch* por exemplo, as suas concepções, linguagens de programação e a estrutura de como se organiza a execução das suas aplicações são distintas. Nas versões mais antigas dos sistemas, as principais diferenças sentidas estavam relacionadas com a falta de qualidade de grafismo do Android, na *framework* de desenvolvimento (normalmente o Eclipse) e nas aplicações em si, e na falha do suporte de *multitasking* no iOS. Hoje esses desafios foram superados e as versões atuais dos sistemas já têm qualidade gráfica de alta resolução e suportam ambas *multitasking*. As principais diferenças ficam assim marcadas nas linguagens de programação das aplicações para cada sistema, os ambientes de desenvolvimento utilizados e os propósitos finais das aplicações. Na plataforma Android o desenvolvimento e venda de aplicações é grátis, ao contrário do iOS que se tem de pagar à Apple Inc. uma licença anual de desenvolvimento e uma taxa sobre a venda de cada aplicação (cerca de 30%). Este facto leva a que o Android tenha uma comunidade de desenvolvedores muito maior e seja utilizado num leque de áreas muito mais vasto, uma das quais a utilização em ambiente académico ou estados embrionários de projetos profissionais. Outro ponto importante, que reforça esta realidade tem a ver com a linguagem de programação utilizada, JAVA para o Android e Objective-C para o iOS, sendo o JAVA mais utilizado no ensino da programação nas academias. Por estas duas razões, com o acréscimo de existir maior facilidade em adquirir dispositivos compatíveis com Android utilizou-se este sistema no projeto do C2S.

Android

O Android é uma plataforma *Open Source*. A ferramenta aconselhada para desenvolvimento de código e dos diferentes interfaces (janelas, imagens, botões, etc.) é o Eclipse, distribuída por um Software Development Kit (SDK) que contém ainda um emulador e ferramentas de análise de performance e debugging. Como todos os sistemas operativos, este possui várias versões, fruto da evolução da tecnologia, estando atualmente a ser desenvolvida a versão 5.0 e a mais utilizada ser a 2.3.X (GingerBread) com 44% da cota de utilizadores. Existem 8 versões do sistema operativo a serem utilizadas ao mesmo tempo, por diferentes utilizadores que possuem diferentes dispositivos. Cada versão do sistema é desenvolvida

internamente pela Google Inc. e só depois disponibilizada para melhorias ou criação de distribuições específicas. As aplicações desenvolvidas terão de ter em conta a versão do sistema a que se destinam, bem como os recursos necessários (periféricos, por exemplo). O SDK disponibiliza ferramentas para cada versão e ainda é possível declarar no ficheiro *manifest*, que condições são necessárias para a instalação e execução da aplicação (versão do sistema, recursos necessários, autorizações do utilizador, etc.), sendo que o utilizador que não reúna essas condições, não consegue instalar a aplicação e em muitos casos, não consegue visualizá-la na loja virtual Google Play.

As aplicações desenvolvidas por terceiros, são o motor de crescimento das plataformas móveis. O resultado do desenvolvimento de uma aplicação é um ficheiro com extensão .apk (Android Package) que contém todos os dados e recursos necessários para a aplicação, e é utilizado para instalação. O sistema Android baseia-se num sistema multiutilizador Linux, em que cada aplicação é um utilizador, sendo que a cada aplicação é-lhe atribuída um ID de utilizador e este, conhecido apenas pelo sistema, é usado para definir permissões de acesso a ficheiros, onde só esse ID pode aceder, note-se que diferentes aplicações podem aceder aos mesmos ficheiros e portanto possuem o mesmo ID. Por defeito, todas as aplicações correm num processo Linux exclusivo para cada uma, sendo que este começa quando um dos componentes da aplicação necessita de ser executado e termina quando já não há tarefas a executar ou o sistema necessita de memória para outra aplicação. Para eficiência do sistemas, existem casos de diferentes aplicações que partilham dados e possuem o mesmo ID, serem executadas no mesmo processo. As aplicações têm ainda a possibilidade de aceder a dispositivos de dados (câmara, cartão SD, contatos, etc.), podendo assim partilhar dados por esses dispositivos.

Portanto o sistema utiliza o princípio do mínimo privilégio, ou *principle of least privilege*, onde cada aplicação só utiliza os componentes e recursos que necessita, limitando assim o acesso a outras partes do sistema e criando um ambiente mais seguro.

Uma aplicação Android é composta essencialmente por blocos, chamados componentes, que se constituem pontos por onde o sistema ou o utilizador pode interagir com a aplicação. Cada componente existe como uma só entidade e desenvolve uma tarefa específica na aplicação, podendo interagir com outros componentes. Existem quatro tipos de componentes, cada um com o seu propósito, função e ciclo de vida, que se podem expor da seguinte forma:

- A Atividade é o componente responsável pelo interface com o utilizador, esta define uma única janela de visualização, capaz de receber *inputs* do utilizador e tomar decisões mediante os seus comandos, podendo mesmo lançar outras atividades (outras janelas) para que se fomente uma experiência de utilização;
- O Serviço é utilizado para execuções de instruções de longa duração ou para atuar sobre processos externos em background. Este não possui um interface de utilizador. Outros componentes, por exemplo uma atividade, podem lançar serviços (ouvir música enquanto se navega na Internet, por exemplo);
- O Fornecedor de Conteúdos, ou melhor o *Content Provider* é o que partilha, disponibiliza, guarda e gere dados de uma aplicação, podendo guardar estes em base de dados e disponibiliza-los a outras aplicações, ou mesmo editar dados de outras aplicações que forneçam um *Content Provider* e permissões de acesso (como aceder à pasta de fotografias da aplicação Câmara). Este permite estabelecer uma partilha de dados, de ações e de experiências entre aplicações;

- O Receptor, ou *Broadcast Receiver* é aquele que recebe e responde a avisos do sistema, como por exemplo bateria fraca, ou envia que descarregou um ficheiro e está disponível para se utilizar. Atuam assim como uma porta, ou *gateway*, da aplicação e portanto dos outros componentes. Este não possui um interface mas pode conter algumas barras de estado ou avisos de notificações.

Um facto interessante no sistema Android, está relacionado com o facto de que qualquer aplicação poder pedir ao sistema para utilizar componentes de outras aplicações para realizar determinada tarefa. Este decide e em caso de permissão, lança o processo e a aplicação solicitada. Por exemplo, se eventualmente na aplicação C2S o utilizador necessita de suporte técnico, pode enviar um e-mail para um departamento técnico, sem que para isso seja necessário construir um serviço de e-mail dentro da aplicação, já que esta pode pedir ao sistema para que os componentes de uma aplicação de e-mail já instalada no sistema (aplicação do Gmail instalada por defeito) realize essa tarefa. Portanto uma aplicação não tem permissão para interagir diretamente com outra, mas o sistema pode servir como mediador. Esta característica cria um design modular, em relação à entidade componente, do sistema e uma grande flexibilidade de cooperação entre ferramentas, que resultam num sistema mais eficiente e iminentemente evolutivo.

Outra característica importante do Android tem a ver com o modo como a conectividade é utilizada, acedendo à tecnologia existente nos *smartphones* (Bluetooth, NFC, USB, etc.). Durante o desenho do projeto questionou-se que tipo de tecnologia se utilizaria para estabelecer a comunicação de registos de portagem entre o OBE e o *smartphone*, concluindo-se que seria interessante utilizar tecnologia NFC por várias razões, descritas na secção seguinte.

NFC

A tecnologia NFC, é um subgrupo das tecnologias Radio-Frequency IDentification (RFID) e foi desenvolvida pela Nokia, Sonny e Phillips em 2004, onde formaram o Fórum NFC. O NFC permite estabelecer comunicações sem fios de curto alcance (cerca de 4 cm), isto é, permite trocar informação entre dois dispositivos, passivos ou ativos, desde que estes estejam próximos o suficiente para se dar um “emparelhamento”. O objectivo é este funcionar de uma forma automática, sem precisar de configurações prévias, tornando a vida dos utilizadores mais fácil.

O NFC foi a tecnologia escolhida para realizar a operação de “leitura” das portagens registadas no OBE. A referida decisão foi tomada entre as três tecnologias, de conexão sem fios, existentes num *smartphone* atual, sendo o Bluetooth, o WiFi e o NFC, tendo em conta a segurança da operação, a experiência de utilização e a inovação da solução. Em primeiro lugar pretendia-se que esta operação fosse segura e que exigisse que o leitor estivesse quão próximo possível do OBE. Portanto, definiu-se que o ideal seria o utilizador/leitor estar dentro do veículo, por forma a que o OBE não fosse acedido por uma entidade totalmente estranha e assim se garantisse privacidade no acesso a dados, bem como identificação de autores de eventuais violações (*hacking*, por exemplo) ao sistema. Uma das soluções tecnológicas poderia passar pela autenticação do utilizador do OBE via palavra-passe, no entanto esta iria aumentar a complexidade do OBE e dificultar a experiência de utilização. Portanto a utilização de uma tecnologia sem fios de “longo alcance”, como o WiFi caiu por terra.

Por conseguinte, considerou-se que o tipo de utilização a realizar seria ocasional, ao fim de uma viagem por exemplo, e que a aplicação móvel não teria de estar constantemente conectada ao OBE, portanto não seria uma transmissão periódica e contínua no tempo, podendo

descartar-se desde logo, uma das tecnologias opoentes, o Bluetooth utilizado para transmissões de tempo real, tipo áudio, e que muitas vezes já se encontra “ocupado” a conectar-se com kits de áudio mãos livres, que exigem emparelhamento. Portanto, o exercício de utilização ganhou contornos e idealiza-se da seguinte forma: depois de algumas horas de viagem e aquando do “abandono” da viatura, o condutor tira o *smartphone* do bolso, encosta-o ao OBE e finalmente, quando eventualmente estiver em casa com uma ligação à Internet rápida e barata, efetua o pagamento das portagens em que incorreu.

Por fim, não menos importante, sabe-se que o NFC é uma tecnologia promissora, por ter as grandes tecnológicas, como o Grupo Samsung, a Nokia, ou a LG por exemplo, no seu desenvolvimento, estando por um lado a difundi-lo pelos seus dispositivos, principalmente *smartphones*, e por outro lado a utilizá-lo para aplicações do dia-a-dia, como os sistemas de pagamento, tal como foi discutido no capítulo do Estado da Arte.

Hoje em dia, fazem parte do Fórum NFC praticamente todos os gigantes tecnológicos (como a HP, a NXP Semiconductors, a Intel Corporation ou a Google Inc. por exemplo) e não tecnológicos (como a American Express ou a Master Card), empenhadas em desenvolver e implementar cada vez mais a tecnologia baseada em NFC. Esta entidade (Fórum NFC) tem desenvolvido especificações baseadas nos standards principais da tecnologia, como o standard ISO/IEC 18092, que define o interface e protocolo de uma comunicação sem fios de dois dispositivos acoplados, a operar nos 13,56 MHz, ou a especificação NFC Data Exchange Format (NDEF) que define o formato de encapsulamento da informação a ser transmitida entre dois dispositivos ou entre um NFC e uma tag, que estejam em conformidade com o Fórum NFC.

O NDEF é o formato de mensagem utilizada para troca de informação entre o OBE e o *smartphone*. Estas mensagens podem conter um ou mais registos NDEF, ou NDEF *records*, cada um transportando um pacote de dados, a dita *payload*, podendo ser associados para transporte de conjuntos de pacotes maiores. Um NDEF record é constituído por três parâmetros que descrevem o pacote de dados, sendo:

- O tamanho do pacote definido no campo PAYLOAD LENGTH, representado por 1 byte para pequenos registos (Short Records, sendo sinalizados com a flag SR do registo) ou 4 bytes para registos normais (Normal Records). Este campo pode ter o valor 0;
- O tipo de dados contidos no registo, para que a aplicação do utilizador possa tomar decisões sob os dados que está a receber, definido no campo TYPE, que recorre ao campo Type Name Format (TNF), com apenas 3 bits, para definir a estrutura do valor do campo TYPE, por exemplo: TNF = 0x02 definindo Media-TYPE e TYPE = texto/plain indica que o pacote de dados transporta texto simples, sem formatações e legível;
- E um identificador opcional para dar informações extras sobre o pacote de dados que poderá ser utilizado pela aplicação utilizadora para identificar mais facilmente um registo de dados que queira aceder;

As mensagens NDEF podem encapsular qualquer tipo de documento. Esta é inicializada com um registo que contém a flag Message Begin (MB) ativada e termina com a flag Message End (ME), como ilustra a Figura 3.6.

O registo NDEF pode ainda conter a Chunk Flag (CF) ativada indicando que os dados estão particionados por diferentes blocos e a flag ID LENGTH (IL) para indicar se o campo



Figura 3.6: Estrutura de uma mensagem NDEF.

ID LENGTH, responsável por indicar o tamanho do campo ID (identificador opcional), está presente no registo e a ocupar um byte ou não. A estrutura e organização dos diversos campos do registo está representado na Figura 3.7.

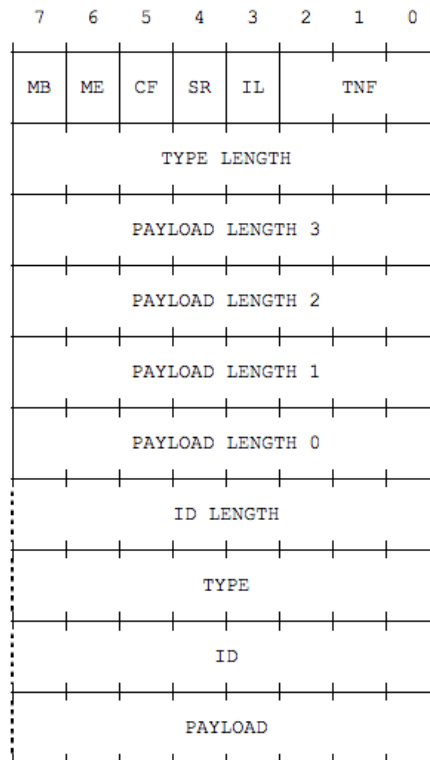


Figura 3.7: Estrutura de um registo NDEF.

Mais detalhes podem ser explorados nos standards e especificações desenvolvidos pelo NFC Fórum.

Voltando ao sistema Android, existe o chamado Android Beam que permite implementar uma comunicação ponto a ponto entre dois dispositivos Android que possuam NFC (mais de 70 modelos até à data [34]). Desde que o dispositivo alvo esteja desbloqueado, quando os dois dispositivos se encostam, uma janela aparece ao utilizador a dizer “Toque para transmitir”, bastando ao utilizador tocar no botão para aceitar a troca de informação. O sistema Android consegue ainda interpretar a mensagem e perguntar ao utilizador que aplicação deseja utilizar, dentro das aptas para ler a mensagem.

Na presente fase do projeto, a tecnologia do dispositivo móvel está baseada em Android, no entanto, no futuro é possível alargar esta aplicação a outras plataformas e projetar alternativas para dispositivos que não possuam tecnologia NFC. Detalhes da implementação são explorados no capítulo de Prova de Conceito.

3.4 Modo de utilização

O sistema C2S é direcionado para o cliente, em praticamente todas as suas vertentes. Principalmente no seu OBE e aplicação móvel que nascem numa filosofia *business-to-consumer* (b2c), procurando ter um interface amigo do utilizador, ou seja, simples e fácil de aceder, depositando um sentido de confiança no utilizador, no que concerne à tomada de iniciativa de leitura e pagamento das portagens, mas não descartando os requisitos mínimos de *enforcement*, necessários à aplicação. O caso de utilização mais simples e regular, começa pelo utilizador/veículo passar por um pórtico (em infraestrutura DSRC) ou uma zona de portagem (em infraestrutura GNSS), o OBE regista as passagens e à posteriori, dentro de um prazo legal (uma semana, no caso Português), o utilizador terá que as descarregar/ler com um *smartphone* e finalmente efetuar o pagamento junto da companhia de portagens via aplicação móvel. No caso da primeira utilização, esta deverá ser realizada, preferencialmente, pelo proprietário do veículo, pelo que este terá de instalar o OBU na própria viatura e descarregar a aplicação móvel para o seu *smartphone*. O C2S tem um modo de utilização simples e baseia-se no princípio de pós-pagamento, para isso é necessário assegurar mecanismos mínimos de fiscalização e de auditoria. As propostas de mecanismos apresentados no âmbito desta dissertação baseiam-se na utilização do sistema ALPR para infraestruturas DSRC, em sistemas semelhantes ao utilizado em Portugal, onde a operação desenrola-se da forma ilustrado na Figura 3.8.

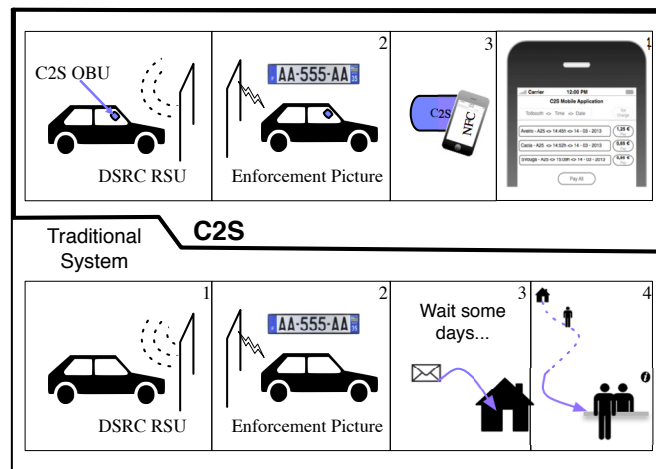


Figura 3.8: Modo de funcionamento do C2S, passo a passo, numa infraestrutura DSRC.

1. Tal como exposto na Figura 3.8, o OBU C2S do utilizador que passa pelo pórtico DSRC, não comunica com o mesmo e portanto o mecanismo de antifraude é acionado. Ao mesmo tempo, o OBU regista internamente a passagem pelo pórtico e instancia a portagem incorrida;
2. O utilizador poderá ler/recolher os registos de portagem do OBU, desde que o veículo não esteja em movimento, por questões de segurança;
3. Dentro do prazo legal, o utilizador tem de pagar as portagens, via aplicação móvel, comunicando às entidades responsáveis (neste fase, a companhia de portagens) que

utilizou o sistema C2S e portanto apresenta prova de pagamento, sendo que o mecanismo de antifraude, de identificação de matrícula, vulgo ALPR, anteriormente despoletado, poderá ser interrompido ou então anulado se já tinha instanciado a infracção.

Este modo de funcionamento é inspirado no processo já existente, onde um utilizador que não possui um OBU DSRC e portanto não comunica com o pórtico aquando uma passagem, tem de se deslocar a um balcão oficial (aos CTT por exemplo) para poder efetuar um pagamento “manual” das portagens e assim impedir o seguimento do processo para uma aplicação de taxa coerciva ao proprietário da viatura, ou seja, uma coima.

No caso da infraestruturas GNSS-GSM, o modelo de *enforcement* ainda não se encontra fechado, podendo passar pela adição ao OBE, de um módulo DSRC que comunique às autoridades que existem registos internos de portagens por pagar. Ou seja, o utilizador não efetuando o pagamento das portagens junto do *back-office*, não consegue eliminar os *logs* das mesmas no OBE, podendo este sinalizar que existem portagens (com mais de uma semana, por exemplo) por pagar. Este modelo assenta no mecanismo de *enforcement* implementado nos sistemas GNSS-GSM, para despistagem de OBUs desligados ou com possíveis anomalias, em veículos em circulação.

3.5 User Cases

Os cenários de utilização são muitas vezes os principais catalisadores na idealização de um produto ou solução, através destes, os desenvolvedores são capazes de identificar oportunidades e desafios de uma forma mais natural. O projeto C2S, não foi diferente. Durante a planificação e desenho do sistema, a linha condutora que se seguiu, baseou-se sempre na solução que esta aplicação pretendia resolver (criação de um interoperável, simples de usar, baseado num interface gráfico) e na experiência de utilização. Logo desde cedo se apontou o EETS, como um caso de estudo onde se poderia aplicar o C2S.

3.5.1 EETS

Uma das restrições mais incompreendidas dos sistemas ETC, da perspectiva do utilizador, é a falha na interoperabilidade dos sistemas de diferentes países, tal como discutido no capítulo do Estado da Arte. Na Europa a maior parte dos países utiliza sistemas baseados na tecnologia DSRC, à exceção da Alemanha e Suíça que utilizam sistemas GNSS-GSM. Na Europa não existe nenhuma evidência de interoperabilidade entre sistemas. Se um condutor pretende viajar de Paris até Aveiro, este tem de utilizar sistemas de pagamento de portagens manuais (quando são existentes) ou viajar em estradas secundárias, ditas nacionais. No pior dos casos, um condutor que tenha de realizar um trajeto internacional frequentemente, terá de colocar um OBU compatível com cada um dos sistemas de pagamento automáticos por onde passa.

Do ponto de vista económico, a rede rodoviária é essencial para a promoção da competitividade, da sustentabilidade e do sucesso dos mercados, pois continua a ser o meio mais utilizado para o transporte de mercadorias a nível Europeu. Desta forma a Comissão Europeia decidiu criar o EETS, lançando a Diretiva 2004/52/CE. Os principais objetivos deste projeto são estabelecer a interoperabilidade e reduzir a proliferação de sistemas incompatíveis na União Europeia. O projeto tem em conta a tecnologia já implementada e como objetivo criar um único dispositivo (OBU) operável nos sistemas existentes.

Tal como discutido no capítulo do Estado da Arte, o EETS envolve vários desafios, um dos quais o tecnológico. O sistema C2S ao implementar uma comunicação direta com um *back-office*, que poderá estar alojado num *EETS provider*, permite flexibilizar processos de comunicação entre tecnologias e sistemas heterogéneos. Mais, com a componente do *smartphone*, a comunicação/declaração de portagens incorridas, é inicializada pelo utilizador/cliente, o que reverte as opções de custos das comunicações para o utilizador/cliente, mediante a tecnologia que utilize (GPRS ou 3G em Roaming, ou WiFi, por exemplo) e por outro lado proporciona uma maior liberdade em custos operacionais para o *EETS Provider* ou os *Toll Chargers*. Embora a complexidade cresça no que toca a mecanismos de *enforcement*, o modelo de funcionamento do C2S, associado às tecnologias inerentes, poderão aplicar-se ao EETS, trazendo vantagens para as entidades envolvidas.

3.5.2 Aluguer de viaturas ou *Car Sharing*

Outra questão, ou falha, dos sistemas atuais surge quando um cliente aluga uma viatura ou usa um serviço *car sharing* (em um veículo sem um OBU) e passa, muitas vezes inadvertidamente, numa infraestrutura ORT. O veículo não comunicando com o pórtico DSRC, é imediatamente acionado o sistema de *enforcement* e são recolhidas imagens do mesmo, sendo este identificado pelo sistema ALPR. No caso do cliente não identificar a referida passagem e comunicá-la à empresa de aluguer do veículo, esta vai ser taxada coercivamente. Desta forma, a empresa de aluguer debita a referida taxa, na conta de crédito associada ao cliente, que finalizou o serviço, muitas vezes, à bastante tempo, que leva à insatisfação deste e à perda de clientes por parte das empresas. Em situações de visitas turísticas ou viagens de negócio de curta duração, o caso torna-se mais grave, pois o cliente abandonou o país e torna-se difícil esclarecer a situação. O cenário menos negativo, acontece quando a empresa de aluguer se apercebe das passagens e notifica o cliente para o pagamento em balcões acreditados (CTT, por exemplo), que embora seja inconveniente constitui uma solução, o que levou as empresas (no caso Português) a começarem a avisar os clientes para que estes "entreguem" as viaturas 48 (tempo de processamento/publicação dos registos de passagem) a 72 horas, antes de abandonarem o país.

No *car sharing* mais propriamente, os problemas podem aumentar, visto que os períodos de aluguer são curtos e a quantidade de utilizadores por viatura elevada. Cabe a cada utilizador, depois da sua utilização, efetuar o pagamento das portagens. Neste segmento, a necessidade de informação sobre as portagens incorridas por cada utilizador é fundamental. O sistema atual causa problemas a muitos clientes, que no caso de clientes estrangeiros não entendem a informação e dirigem-se para infraestruturas ORT, incorrendo em portagens. Por outro lado, causa problemas às empresas de aluguer, que se vêm obrigadas a instalar OBUs nos seus veículos, sem que isso se constitua uma solução, e além disso têm custos operacionais na monitorização das viaturas, bem como na resolução de possíveis conflitos.

Com o sistema C2S o cliente e a empresa de aluguer, têm acesso a um registo de portagens incorridas durante o período de aluguer, apresentando assim uma proposta de solução aos problemas identificados. Esta solução é especialmente direcionada para o segmento do *car sharing*, onde no final de cada viagem/utilização, o cliente poderá pagar portagens, de forma cómoda, e comprovar ao mesmo tempo comprovar à empresa de aluguer. Algumas empresas de aluguer de viaturas já tiram partido das novas tecnologias, como as aplicações móveis (para o registo e aluguer de uma viatura) e o NFC (como chave da viatura), representando assim um caso de utilização ideal para o sistema C2S.

Capítulo 4

Prova de Conceito

4.1 Estrutura e Desenho do Sistema

O sistema C2S nasce numa filosofia de ajudar os utilizadores de portagens eletrónicas, principalmente de infraestruturas ORT, a consultar as suas passagens e efetuar o respetivo pagamento de um modo simples e rápido. Tal como referido no capítulo anterior, o objectivo é projectar um sistema que seja amigo do utilizador, que crie condições para o registo de portagens de condutores que utilizem veículos dos quais não são proprietários e que de certa forma se possa encaixar no desenho do sistema EETS.

A presente secção relata a implementação efetuada da arquitetura do sistema, exposta no capítulo anterior. Portanto dividiu-se o sistema em três grandes blocos: o OBE fixado na viatura; o *smartphone* como elemento de interface com o utilizador e principal módulo de comunicação; e o *back-office*, centro controlo da aplicação móvel e gestor das portagens incorridas. No âmbito desta dissertação, procurou-se desenvolver os sistemas mais próximos do *front-end* do sistema, ou seja, o OBE e a aplicação móvel do *smartphone*, para se poder criar uma prova de conceito, de forma a se poder simular a experiência de utilização. Numa fase seguinte do projeto será importante desenvolver o *back-end*, perto do *back-office* e das entidades gestoras do mesmo, para que se complete toda a linha do sistema. É ainda importante referir que a estrutura desenhada é o mais conceptualmente e funcionalmente flexível para se poder adaptar a possíveis mecanismos ou regras impostas pelo *back-end*.

Tal como referido, a prova de conceito do OBE foi desenvolvida utilizando a plataforma Raspberry Pi com a distribuição Arch Linux, devido às suas características computacionais e dimensionais, em conjunto com outros periféricos, como o receptor GPS, o módulo NFC e eletrónica extra (interruptores e LEDs de sinalização).

4.1.1 *On Board Equipment*

O OBE comporta-se como uma extensão de um *data logger*, isto é, comporta-se como um módulo provido de inteligência para interpretar os dados que lê (coordenadas, data, hora, número de satélites, etc.), registando os que correspondem a zonas de portagem e disponibilizando estes para transmissão via NFC. Para tal decidiu-se dividir este sistema em três blocos, tal como ilustra a Figura 4.1, sendo o primeiro responsável por aquisição e disponibilização dos dados, denominado de *Sniffer*; o segundo bloco é responsável pela comparação dos dados recolhidos com a base de dados interna, aferição e registo dos casos em que exista

correspondência, denominado de *Matcher*; e finalmente, o terceiro bloco, denominado de *Logger*, é o responsável pela gestão de persistência dos dados em registos locais, nomeadamente gestão dos registos de portagens incorridos e dos registos já declarados como pagos, bem como transmissão destes para o módulo NFC.

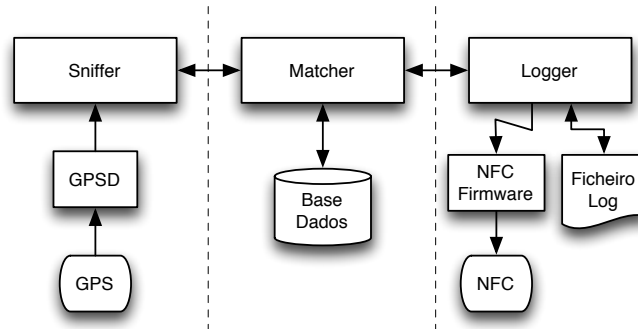


Figura 4.1: Estrutura organizacional do *On Board Equipment*.

Decorre da estrutura apresentada, a existência de três processos concorrentes, onde os que estão mais próximos da leitura de dados do receptor GPS têm prioridade. O motivo para organizar a estrutura desta forma, nasceu do facto da leitura de dados e a pesquisa destes na base de dados (portanto a comparação do processo *Matcher*) ter de ser independente, para que no caso de uma pesquisa mais demorada, a leitura de dados do GPS não fique inviabilizada. Isto é, os dados são recebidos e capturados pelos *Sniffer*, que fica encarregue de enviá-los para o *socket* onde estes ficam armazenados, sob o formato de mensagens. O processo *Matcher* atende e trata cada mensagem, tentando encontrar uma correspondência na base de dados, tendo esta operação uma latência proporcional à complexidade e ao tamanho da base de dados local. Ao mesmo tempo é necessário um terceiro processo, o *Logger*, que coordena o módulo NFC responsável por atender pedidos do utilizador quando este tenta comunicar com o OBE, ou seja, quando o utilizador “encosta” um *smartphone* ao NFC este deve responder com o registo de portagens incorridas e deve transmitir ao sistema (OBE) os registos das portagens já pagas. Esta operação é totalmente assíncrona e ocasional, ocorrendo por livre iniciativa do utilizador, daí a importância da existência de um processo exclusivo para esta função. Por outro lado, o facto de se dividir o sistema em três processos independentes, ganha-se modularidade, organização, e flexibilidade numa eventual integração de novas ferramentas ou utilização destes processos noutros projetos. Em contrapartida, o facto de se utilizar processos e não *threads* faz com que exista um maior esforço computacional, devido à constante mudança de contexto do processador.

Leitura de dados - *Sniffer*

A leitura dos dados representa o *front-end* deste módulo e é feita através de um receptor GPS, que se conecta ao RPi via *serial USB*, ou seja, atuando como uma comunicação série, a 4800 de Baudrate, 8 bits de dados, sem bit de paridade e 1 stop bit. O receptor GPS utilizado foi o BU-353 da GlobalStat [35], que possui um chip SirfStar III, opera com uma frequência L1 nos 1575,42 MHz, e 20 canais, possui um sensibilidade de -159 dBm e por defeito utiliza o protocolo NMEA 0183 [36] para comunicar os seus dados. Este ainda tem suporte

para as plataformas de GPS assistido EGNOS e Wide Area Augmentation System (WAAS), bem como tempos de aquisição relativamente baixos, consumos de potência abaixo do 1 W (importante para o RPi) e precisões de 10 metros no pior dos casos (segundo o *datasheet* do dispositivo [35]).

Do lado do RPi, ou seja, do sistema computacional *host* deste periférico, é executado um serviço, no sistema Arch Linux, responsável por receber, organizar e disponibilizar os dados do receptor GPS. O programa responsável por esta tarefa é um serviço Disk And Execution MONitor (daemon) denominado GPSd. Um daemon é um programa que é executado em *background* e que não depende da intervenção do utilizador, ou seja, toma decisões autónomas e independentes. O GPSd configura a porta série, com as especificações do fabricante referidas anteriormente, recebendo os dados no formato do protocolo NMEA 0183 e posteriormente procede ao *parsing* dos mesmos, simplificando-os e disponibilizando-os em formato de estrutura de dados. Este liga-se ainda a uma biblioteca em C, oferecendo assim ferramentas de programação, ou seja, funções capazes de disponibilizar os dados internamente aos programas desenvolvidos. O GPSd é utilizado para monitorizar os GPS dos sistemas Android (a partir da *release* 4.0) e portanto é considerado *software* de qualidade, extremamente auditado, sendo premiado pela Alliance for Code Excellence [37].

Portanto o GPSd é o serviço responsável pela gestão do receptor GPS e disponibiliza os dados recebidos para o sistema computacional sob a forma de *socket*, neste caso um *socket* local, ou seja, apenas para comunicação entre processos, internos ao sistema computacional. Na subsecção seguinte é explicado mais ao pormenor o conceito do *socket*, pois a comunicação entre os blocos existentes, ou seja, entre os processos em funcionamento, está baseada neste tipo de interface de comunicação.

A rotina desenvolvida, intitulada *Sniffer*, tem como principais objetivos configurar e abrir o *socket*, que actua como *server* do cliente GPSd, aferindo a disponibilidade dos dados, efetuando a sua leitura e finalmente enviando para o módulo *Matcher*, recorrendo também a um *socket* cliente. Esta rotina divide-se em várias funções, sendo as primeiras de configuração dos *sockets* necessários e as restantes as de leitura e interpretação dos dados, tal como ilustra o fluxograma da Figura 4.2. Note-se que durante a execução, o processo está em ciclo infinito, só abandonando o ciclo com o despoletar da interrupção gerada pelo botão stop disponível ao utilizador/*tester*, para reiniciar o sistema no caso de acontecer algum bloqueio inesperado. É importante assinalar que estando este processo a operar no ambiente do sistema operativo e embora tenha uma prioridade de execução elevada, o sistema tem poder e meios para "matá-lo", sendo necessário efetuar um *reset* à máquina de estados. Portanto o termo ciclo infinito adequa-se a situações de teste e de prova de conceito, num ambiente controlado, limitado ao âmbito deste projeto.

Durante a execução, a função "GPS Waiting" espera por uma mensagem no *socket*, tendo um *timeout* predefinido para libertar o processador, por exemplo 100 ms. Logo de seguida, ocorrem algumas validações quanto à consistência dos dados e uma comparação com os dados passados em que: se as coordenadas recebidas são semelhantes ao longo do tempo, ou seja, o veículo não se movimentou mais do que um *offset* predefinido (10 metros, por exemplo), não é reportada uma nova mensagem para o *socket* cliente do processo *Matcher*.

Finalmente, observa-se a *flag* associada ao botão stop, ativada numa rotina de interrupção criada para o efeito, e em caso negativo o ciclo repete-se. Existe ainda um pequeno interface, para o caso de o utilizador, ou melhor *tester*, aceder à linha de comandos, poder configurar o tipo de dados que pretende receber: de um receptor GPS real para testes *in loco*; ou de um "emulador", como o GPS_Fake, programa que se liga ao GPSd transmitindo-lhe mensagens

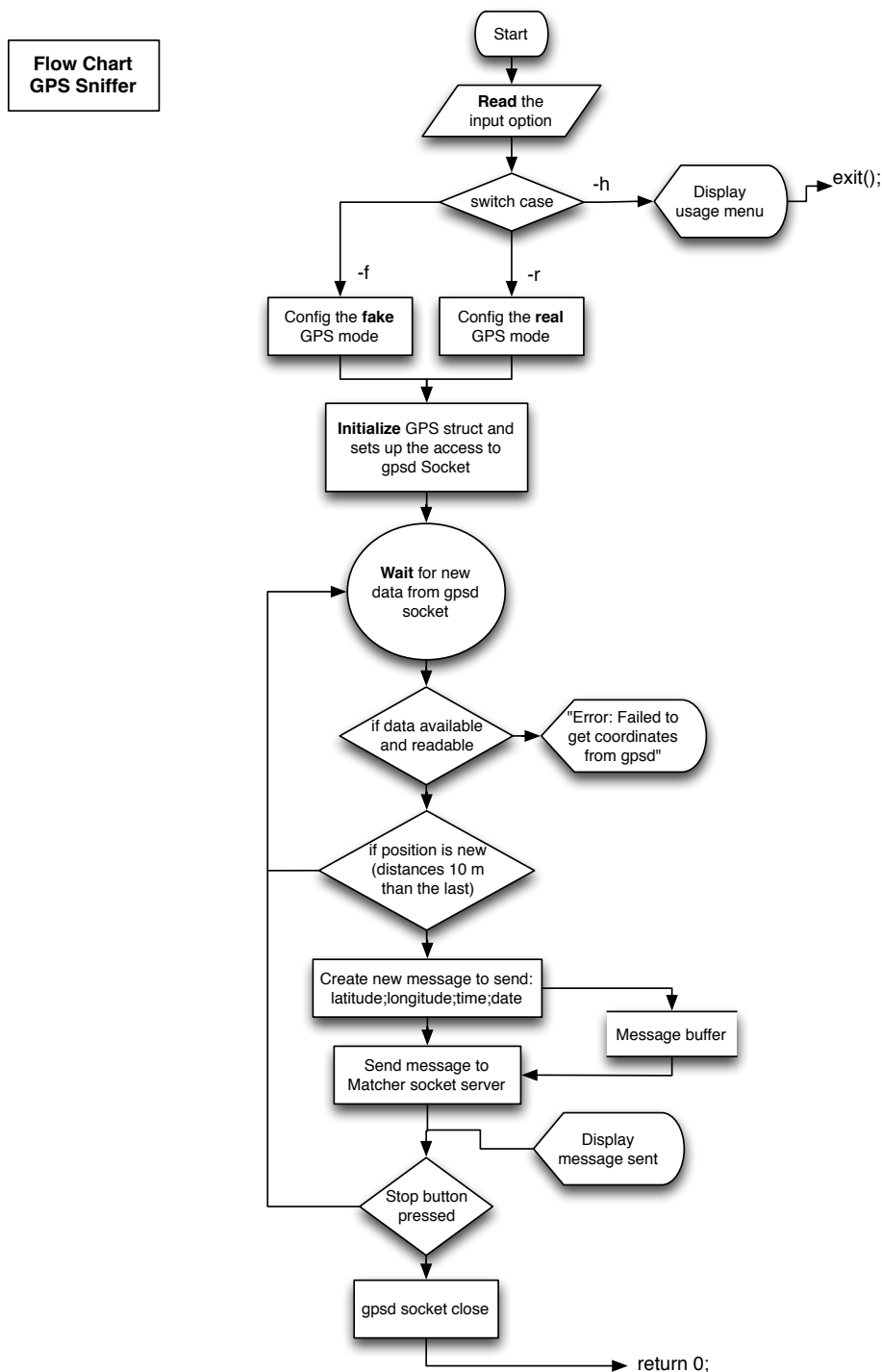


Figura 4.2: Fluxograma da rotina do módulo *Sniffer*.

NMEA de um ficheiro *log* carregado pelo *tester*, ideal para simulações no laboratório; e por outro lado, o *tester* poder observar que mensagens estão a ser enviadas para o módulo a jusante.

Comunicação entre processos por meio de *Sockets*

Um *socket* é um ponto de comunicação que pode ter um nome e um endereço numa rede. Os processos que utilizem *sockets* podem estar a ser executados no mesmo sistema (na mesma máquina) ou operar em rede, desta forma consegue-se distribuir tarefas por diferentes máquinas e assim tornar o sistema mais eficiente. Os *sockets* são muitas vezes utilizados para comunicações numa rede constituída por diferentes máquinas e essa é uma das razões para a utilização dos *sockets* nos interfaces de comunicação entre cada processo do OBE. Assim tem-se uma abordagem modular e abre-se a possibilidade da utilização destes em sistemas que funcionem em rede dentro do veículo e que eventualmente necessitem das funcionalidades de um dos processos implementados, por exemplo articular a navegação GPS do utilizador com o processo *Sniffer*.

Os *sockets* utilizados para comunicação entre processos do mesmo sistema operativo são denominados Unix *domain sockets* ou IPC *sockets* (*inter-process communication*) e assimilam-se aos *pipes*. Estes são bidireccionais e estabelecem uma comunicação entre uma entidade receptor, chamada *server* e uma emissora, chamada cliente. Os Unix *domain sockets* utilizam o próprio sistema de ficheiros do sistema operativo para endereçar o ficheiro (o caminho até ao ficheiro: `/root/comm/cliente` por exemplo). No sistema os *sockets* são interpretados como *inodes* ou *index nodes*, ou seja uma estrutura de dados que armazena informação sobre um objecto do sistema de ficheiros, neste caso *socket*, não contendo dados, e assim permitindo o acesso de dois processos ao mesmo *socket*.

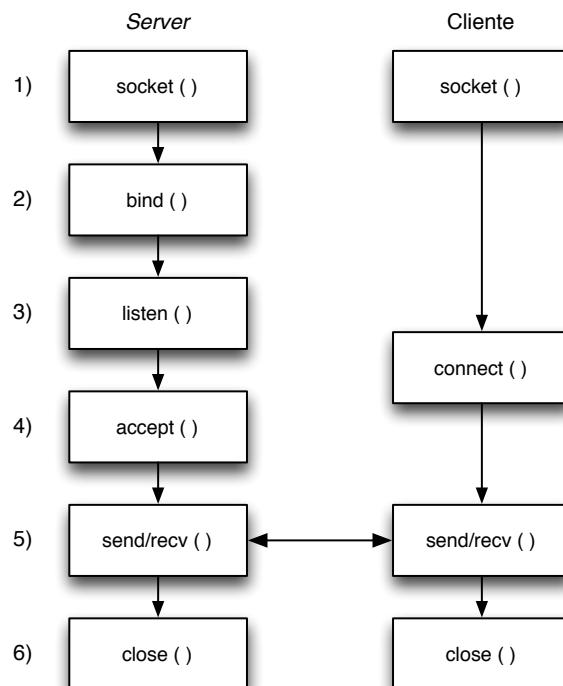


Figura 4.3: Fluxograma da criação de um *socket*, baseado em [6].

Para implementar um *socket* é necessário utilizar um conjunto de rotinas configuradas e ordenadas de uma forma específica [6], tal como ilustra o fluxograma da Figura 4.3. Para criação dos pontos de comunicação e obter o *socket descriptor* chama-se inicialmente a rotina

"socket". De seguida, do lado do *server* é executado a rotina "bind" que irá proceder à conexão do socket ao endereço no sistema de ficheiros do sistema e fica em escuta, com a rotina "listen", até que receba uma mensagem do cliente. O cliente, através da rotina "connect" questiona o *server* para estabelecer a comunicação e este utiliza a rotina "accept" para aceitar o pedido. Quando a conexão é estabelecida as funções "send" e "receive" responsabilizam-se pela troca de mensagens, e quando se pretende cessar comunicações o cliente ou o *server* executam a rotina "close", como se pode observar na Figura 4.3.

Embora a utilização de *sockets* acarrete menores eficiências computacionais, a utilização destes deveu-se aos propósitos já descritos, no entanto é necessário ter em atenção que o facto de estes se tornarem pesados poderão afectar os compromissos temporais exigidos, em máquinas mais lentas.

Processamento de dados - *Matcher*

A rotina *Matcher*, muito sucintamente, tem como principais funções: recepcionar mensagens do processo *Sniffer*; aferir se as coordenadas da mensagem recebida correspondem a posições de zona de portagem, realizando uma pesquisa na base de dados; e finalmente enviar um identificador da portagem incorrida para o processo *Logger*.

O papel mais importante da rotina em questão, reside na aferição de passagem em portagem. O método utilizado para o efeito consiste em guardar as coordenadas (2 dimensões, latitude e longitude) de duas mensagens consecutivas, definir um vector de posições entre os dois pontos e realizar uma pesquisa na base de dados de cada uma das posições. Este método foi criado para colmatar o facto do tempo de refrescamento do receptor de GPS ser 1 segundo, portanto no caso de um veículo a 120 km/h (velocidade perfeitamente admissível) a distância percorrida durante 1 segundo ronda os 33,3 metros e no caso de um veículo a circular a 200 km/h (velocidade legal em alguns países da UE) a distância percorrida ronda os 55,5 metros. Desta forma a comparação entre as duas posições (a posição do pórtico na base de dados e a posição do veículo) não é feita diretamente, ou seja, no plano da longitude e latitude, em vez de se simplesmente calcular a distância entre dois pontos é a calculada a distância entre um ponto e um conjunto de pontos pertencente ao segmento de recta definido por duas posições consecutivas do veículo. A utilização deste modelo implica algumas reservas, fruto da sua debilidade.

Este modelo de detecção de passagem em determinada zona (zona de portagem), suscita duas questões que poderão tornar-se problemáticas se não forem estabelecidos certos compromissos. A primeira questão prende-se na geometria do traçado das vias rápidas, que maioritariamente possui rectas e curvas bastante suaves, mas em certos trechos poderão estar curvas de transição, como as clotóides (as mais utilizadas), que são curvas suaves e com raios de curvatura progressivamente mais curtos, ou seja progressivamente mais acentuadas, onde poderão existir zonas de portagem. A velocidades elevadas, os veículos percorrem distâncias que rondam os 50 metros de estrada durante 1 segundo (taxa de refrescamento do receptor), portanto serão definidos vectores que seccionam (ou "cortam") as curvas de transição e para o pior caso poderá ocorrer uma falha na detecção, tal como ilustra a Figura 4.4. Uma das soluções a aplicar poderá ser a definição de zona de portagem numa área maior sob pena da ocorrência de falsas detecções de estradas paralelas. Outra solução poderá passar pela avaliação dos traçados das vias confrontando a localização das zonas de portagem com a distância máxima ao vector de posições numa curva acentuada (onde se realiza o "corte" das curvas), podendo até modificar-se a geometria da área de portagem passando de um simples círculo

para uma elipse, ou com mais complexidade, um rectângulo.

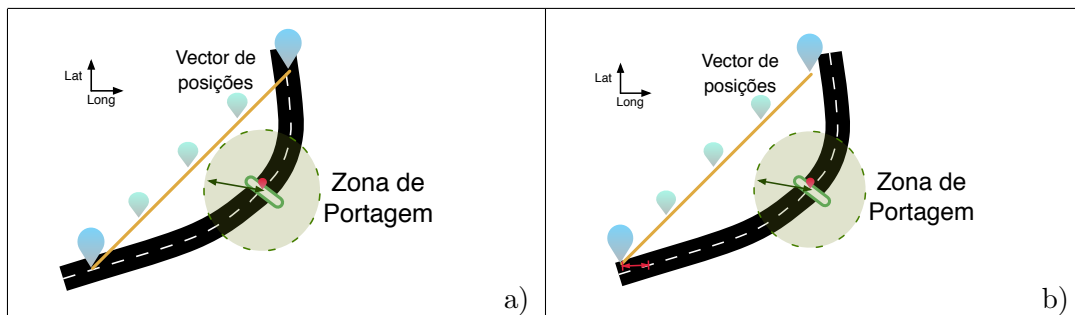


Figura 4.4: Falha de detecção em curvas de transição: a) com ausência de erro; b) com erro na longitude.

A primeira solução é mais simples de implementar visto que necessita apenas de um ponto e um raio para definir uma área ou zona de portagem, enquanto que a segunda é mais robusta mas mais complexa de implementar, portanto computacionalmente mais exigente pois envolve mais pontos na definição da sua área e consequentemente mais comparações de coordenadas a realizar. Neste momento em operação, para efeitos de prova de conceito está-se a utilizar uma abordagem onde se definiu uma zona de portagem com um raio de 100 metros e um vector com posições distadas de 10 metros.

A segunda questão reside no facto da memória do sistema ter de ser contextualizada, isto é, a última posição do veículo (gravada em memória) comparada com a atual terá de ser validada quanto à distância e data, sob pena de se estar a traçar um vector entre duas posições muito distantes perdendo-se consequentemente eficiência computacional. O sistema utilizado realiza facilmente operações aritméticas, como a de calcular a distância geográfica entre dois pontos, tendo em conta a curvatura da Terra, e portanto pode-se utilizar esta operação para verificar se os pontos estão suficientemente perto, para viabilizar o processo de traçar o vector de posições. No caso do sistema a utilizar não ter apetências para desenvolver este tipo de cálculos, poderá-se focar a validação na data e tempo do registo das posições, pois posições desatualizadas têm uma maior probabilidade de estarem geograficamente distantes.

Além dos cuidados a ter na implementação do método proposto, é importante ter presente que existe uma debilidade, no mecanismo de detecção exposto, relacionada com a falsa detecção de veículos em infraestruturas paralelas, como por exemplo autoestradas com os dois sentidos contíguos, onde se torna difícil identificar em que sentido o veículo se desloca. Mais, este problema agrava-se com o erro associado ao GPS, na ordem dos 10 metros (distância bastante superior à separação das vias). Para colmatar este problema poderá aplicar-se duas soluções, uma com a definição de uma única zona de portagem para os dois sentidos de uma via rápida sob pena de as características (preço, por exemplo) serem iguais e portanto não poderem ser diferenciadas. Enquanto que a segunda solução, garante a separação de zonas de portagem em duas vias contíguas, baseando-se no desfasamento das duas zonas, de alguns metros, em vias contíguas com sentidos opostos, tal como ilustra a Figura 4.5.

Em alguns casos, torna-se necessário criar zonas secundárias de passagem, denominadas *checkpoints* ou zonas de validação, que se articulam com as zonas de portagem principais, que são vizinhas de outras infraestruturas (vias rápidas, viadutos, estradas nacionais), e onde é necessário efetuar o despiste de falsas detecções, originadas pela passagem de veículos nas vizinhanças. Os *checkpoints* são colocados depois das zonas principais de portagem, no sentido

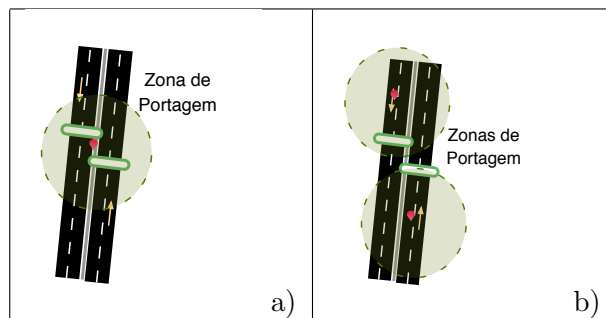


Figura 4.5: Proposta de localização de zonas de portagem: a) uma única zona; b) zonas desfasadas.

de circulação, distanciados algumas dezenas de metros destas, de forma a que se localizem em áreas isoladas de vias nas vizinhanças, e assim quando um veículo intersecta a zona de portagem principal e seguidamente o seu *checkpoint*, assume-se que o veículo incorreu na portagem.

Depois da confirmação da passagem em zona de portagem, é criada uma mensagem com o identificador (ID) da portagem, hora e data, e é enviada para o *socket server* do processo *Logger*, onde será registada. Portanto, no geral a rotina começa por configurar os *sockets* que utiliza, o *server* que recebe as mensagens provenientes do processo *Sniffer* e o cliente por onde se envia mensagens para o processo *Logger*, bem como configuração dos portos GPIO para controlo de um *buzzer*, que é ativado a quando da passagem por zona de portagem, e botões de controlo/teste (botão de stop), como exposto no fluxograma da Figura 4.6. De seguida é carregada a base de dados em memória e são impressas mensagens de confirmação/consistência/conclusão no terminal para o *tester*.

Logo após as configurações iniciais, dá-se início ao ciclo repetitivo que espera pela recepção uma mensagem com as coordenadas do veículo, valida se entre a posição anterior e a atual ocorreu alguma intersecção com zonas de portagem, e no caso afirmativo o *buzzer* é ativado e uma mensagem é enviada ao processo *Logger*, tal como ilustra a Figura 4.6. Este ciclo repete-se, até que o botão stop seja pressionado pelo *tester*. Nesta versão de prova de conceito do sistema é efetuado os logs das operações que se estão a realizar e por outro lado existem dois parâmetros de entrada, que servem para dar entrada do endereço do ficheiro que contém a base de dados com as zonas de portagem, e para ajustar uma variável, denominada *range*, que determina o raio da área das zonas de portagem.

Registo e envio de dados - *Logger*

A rotina *Logger* foi desenhada com o intuito de registar e gerir os registos de portagens em memória estática, tal como o nome indica. Além disso, esta rotina tem uma extensão nas suas funcionalidades, atribuindo-se a tarefa de comunicar com o módulo NFC, para onde envia o registo de portagens incorridas e recebe os registos das portagens pagas pelo cliente na aplicação móvel, via porta série, nomeadamente *USB série*. Na presente fase do projeto, utilizou-se como módulo NFC, um *kit* Arduino Uno [38], com um *shield* NFC, a versão 1.0 do Seeed Studio [39], visto ser um *kit* já adquirido no passado, com um paradigma e uma comunidade *open source*, sob o qual estão-se a realizar os primeiros passos nas comunicações Peer-to-Peer (P2P) sendo um dos pares um *smartphone*. Portanto o *shield* comunica com o

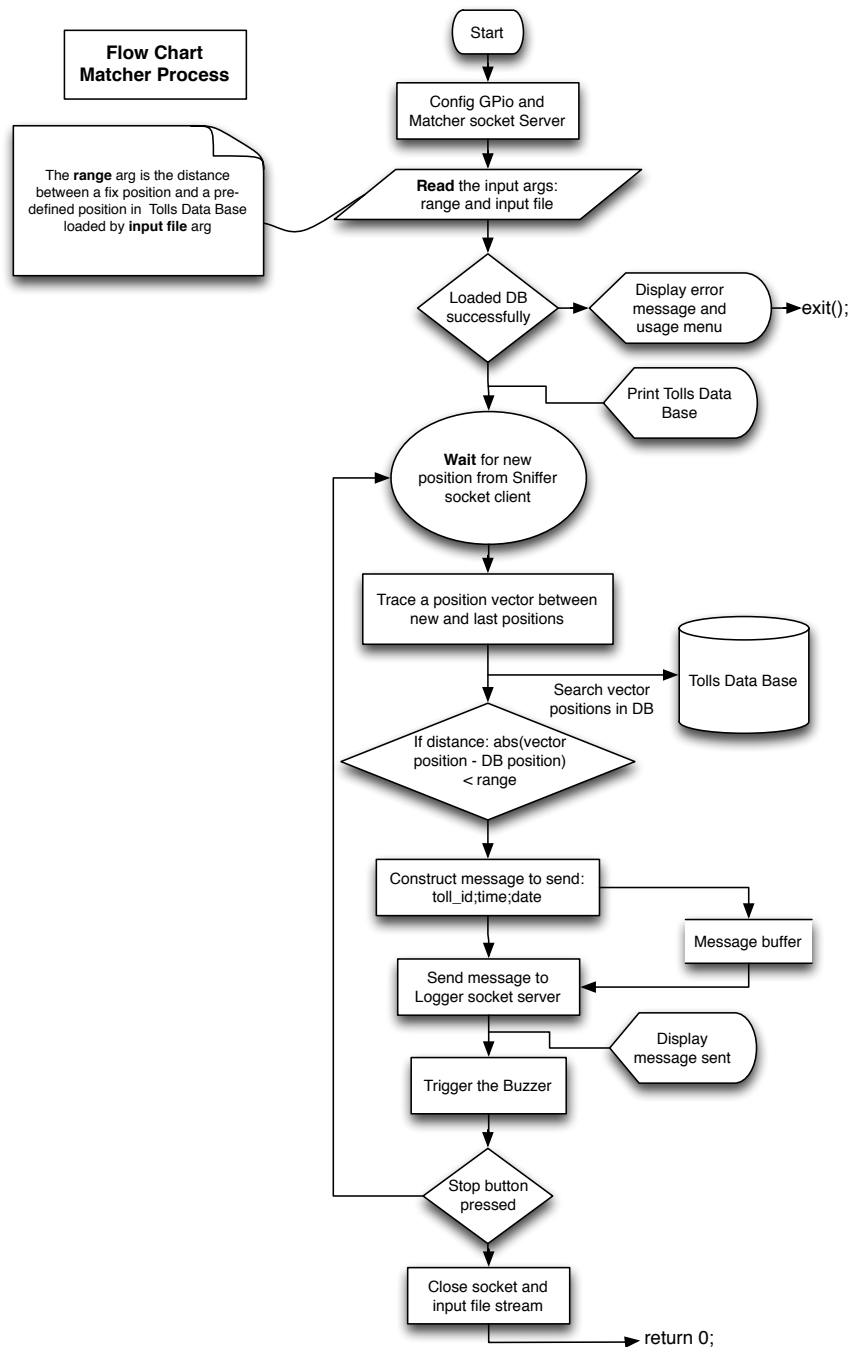


Figura 4.6: Fluxograma da rotina do módulo *Matcher*.

Arduino por SPI e para este foi desenvolvido um *firmware*, baseado nas funções disponibilizadas pela biblioteca PN532 SPI da NXP Semiconductors, que é capaz de ler mensagens provenientes da unidade principal (RPi) via Universal Asynchronous Receiver/Transmitter (UART), constituir uma mensagem e disponibiliza-la no NFC até que um *smartphone* se aproxime e se estabeleça uma comunicação.

Tal como se pode observar no fluxograma da Figura 4.7, o processo *Logger* começa por configurar as comunicações para a porta série e "abrir" o *socket server* onde receberá as mensagens das portagens incorridas provenientes do processo *Matcher*. Mais, lê o ficheiro dos *logs* das portagens incorridas e ainda não pagas, e carrega estes registos em memória. Quando recebe uma nova mensagem, é adicionado um novo registo ao ficheiro de *logs*, de modo a garantir persistência dos dados e é criada uma nova mensagem, com os novos registos, para ser enviado para o módulo NFC. Para comunicar com o módulo definiu-se um protocolo de comunicação, onde existem dois tipos de mensagens enviados do módulo para a unidade principal, sendo os comandos e os dados, onde os primeiros podem sinalizar o envio de dados ou pedir para que se envie novas mensagens. Existem 4 tipos de comandos que sinalizam 4 estados em que o *firmware* do módulo se pode encontrar:

- o **LOOP**, que indica que o módulo já realizou todas as configurações necessárias junto do *shield* NFC e está preparado para comunicar com outro dispositivo, estando à espera no modo de *pooling*;
- o **WAIT**, é enviado depois do módulo não encontrar qualquer dispositivo para estabelecer comunicação e entrar em modo de espera, poupando alguns recursos e despertando quando um dispositivo se tentar acoplar. Ao mesmo tempo, neste estado bem como no anterior (LOOP), o módulo NFC está também "à escuta" da unidade principal, ou seja, encontra-se disponível para receber novas mensagens, sendo estes interrompidos por uma interrupção gerada pela porta série em que a sua rotina de atendimento determina que se reinicie todo o processo com a nova mensagem recebida;
- o **Handshake**, é um comando que vem seguido da mensagem que foi enviado para o módulo, ou seja, funciona como um eco da mensagem enviada e serve para garantir concordância nos dados no módulo e na unidade principal, bem como detectar possíveis erros ocorridos durante a transmissão das mensagens;
- o **PAID**, é um comando seguido de uma mensagem que contém registos de portagens pagas pela aplicação móvel, portanto sinaliza a unidade principal que irá receber um mensagem que terá que guardar, interpretar e tomar decisões perante o seu conteúdo.

O processo *Logger* lê todas as mensagens recebidas na porta série, através do atendimento a uma rotina de serviço à interrupção, tomando decisões mediante o seu conteúdo, sendo os comandos recebidos o fator discriminante. Quando é recebido um Handshake, os dados associados são registados e comparados com os enviados, em caso de haver alguma diferença repete-se o envio das mensagens. Se o comando recebido for PAID, as mensagens com os registos de portagens pagas são gravadas num *buffer* circular e é efetuada uma pesquisa dos registos pagos, uma validação das datas de ocorrência da portagem, e finalmente a atualização do ficheiro de *logs* onde são eliminados os referidos registos. Quando a unidade principal recebe um comando LOOP ou WAIT, compara os registos atuais existentes nos *logs* com a última mensagem do Handshake, para garantir concordância nos dados. Por outro lado, depois de receber algum destes comandos (LOOP ou WAIT) o sistema tem permissão para enviar novas mensagens para o módulo NFC que entretanto foram recebidas do *Matcher*. Seguindo o paradigma dos processos anteriores, o *Logger* recebe como argumento o endereço para o ficheiro de *logs* e imprime mensagens de execução no terminal da linha de comandos. O sistema poderá ser interrompido pelo pressionar do botão stop, tal como se observa no fluxograma da Figura 4.7.

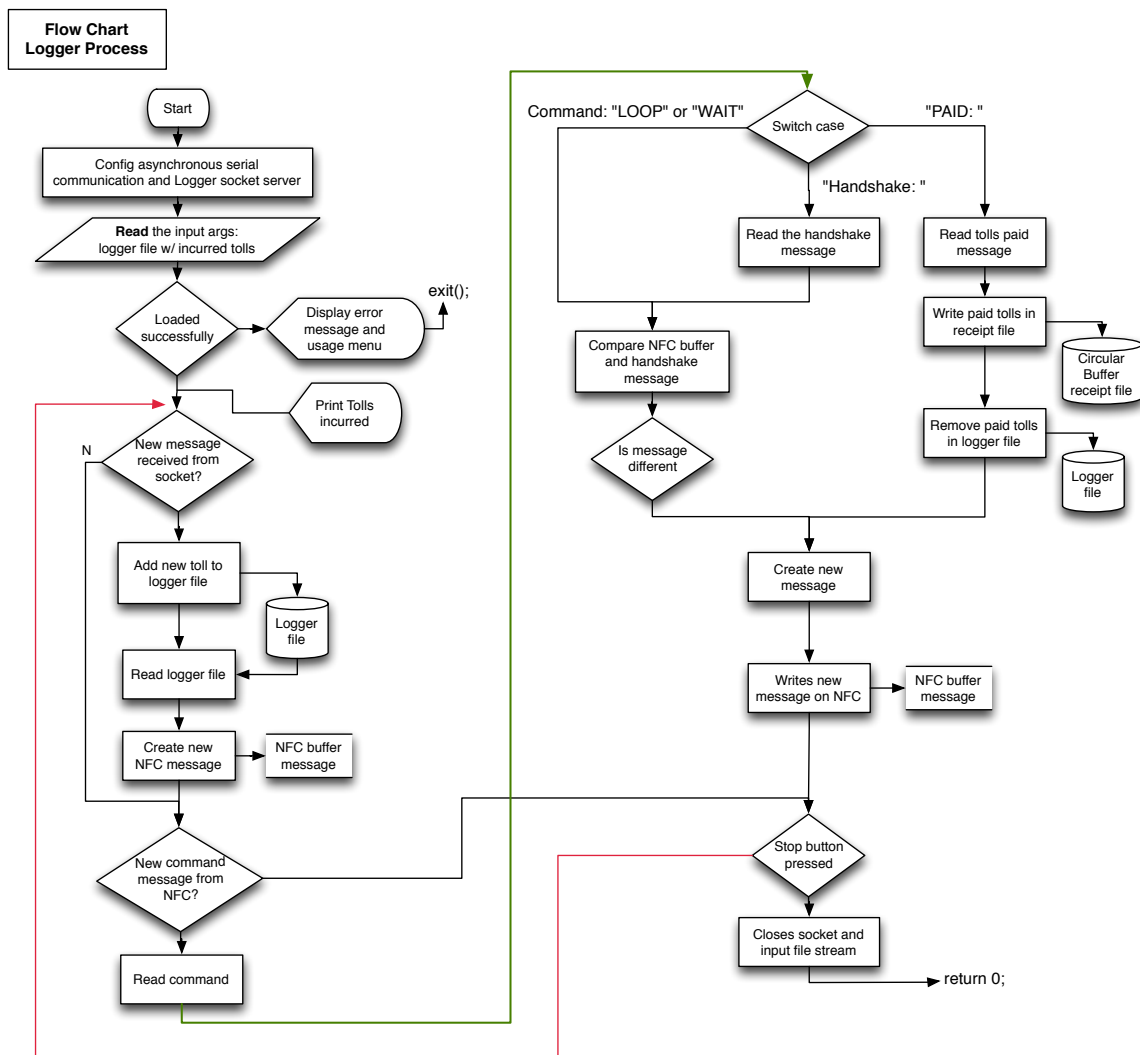


Figura 4.7: Fluxograma da rotina do módulo *Logger*.

Com módulo de comunicações à parte, pretende-se criar um sistema autónomo, que fique responsável pela comunicação com o cliente (aplicação móvel) e portanto dedicado à sua mediação e atendimento, visto que este ocorre com uma frequência completamente ocasional, sendo impossível de prever. Por outro lado, desta forma cria-se uma camada de abstração que permite que a tecnologia, utilizada para as comunicações, possa ser mudada no futuro, por exemplo de NFC para Bluetooth, bastando mudar o módulo de comunicações e manter o mesmo protocolo.

O OBE é assim constituído pelos três processos descritos, cada um desempenhando funções diferentes ao mesmo, mas articulados entre si, podendo se observar o modo como cooperam no fluxograma geral do sistema ilustrado na Figura 4.8.

Como nota final, da implementação do OBE, é importante referir que durante o desenho do sistema, lançou-se a hipótese da utilização de um módulo DSRC para detectar a passagem em pórticos baseados nesta tecnologia. Embora o método de geo-localização seja capaz de

C2S OBE Overview

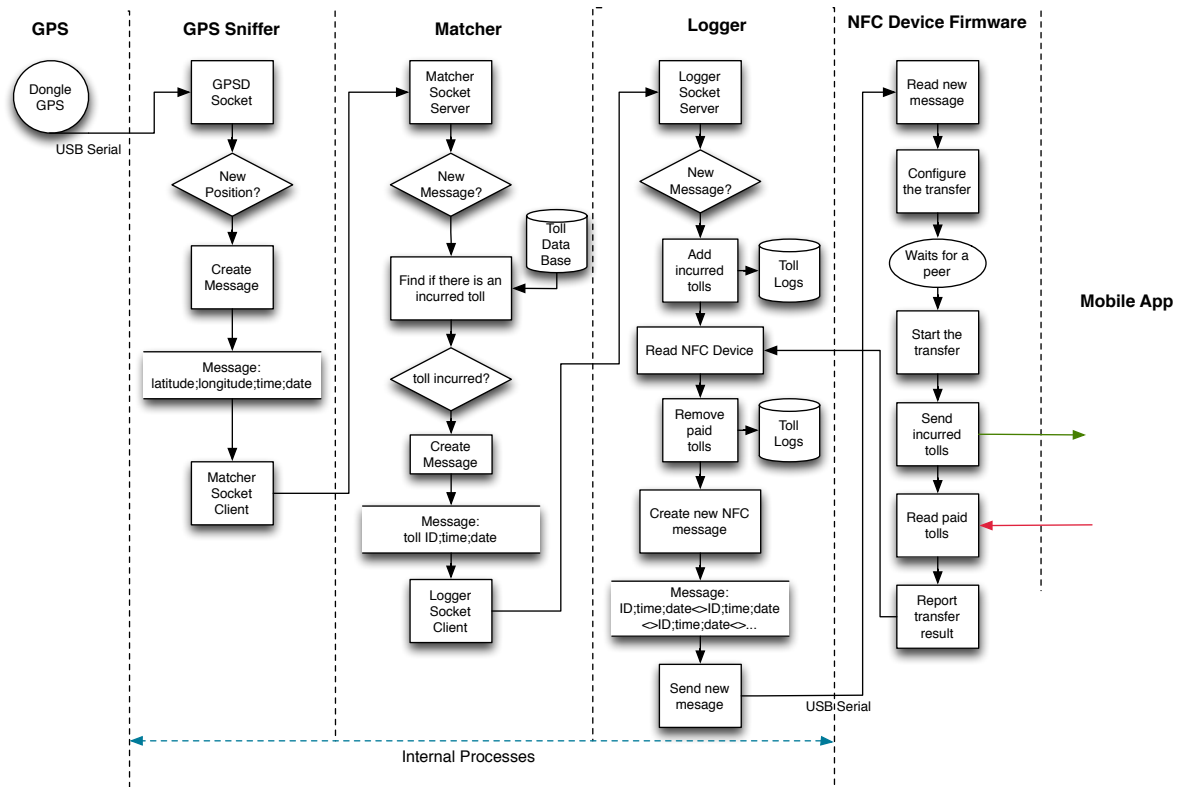


Figura 4.8: Fluxograma geral do OBE.

realizar a função de detecção de zona de pórtico, a comunicação com uma infraestrutura de estrada ou pelo menos a detecção da mesma, poderá ser importante para cenários de validação/confirmação de passagem em determinada localização. Assim, deu-se importância à funcionalidade descrita e propôs-se para a sua implementação, a utilização de um módulo de rádio, presente nos OBUs que operam em DSRC, ligado a uma porta GPIO, com o devido acondicionamento de sinal, de forma a constituir-se uma linha de interrupção à unidade principal, neste caso RPi. Nesta fase do projecto, dado que se pretende uma prova de conceito operacional em laboratório, para os testes desta funcionalidade, utilizou-se um botão conectado ao GPIO para simular detecção de um pórtico DSRC, que gera uma interrupção no processo *Matcher* sendo este obrigado a validar as coordenadas presentes para o registo da portagem referida.

Depois da implementação das rotinas descritas em linguagem C e dos primeiros testes de operação das mesmas, foi necessário avançar para o desenvolvimento da aplicação móvel para se conseguir tirar conclusões fidedignas sobre a resposta do sistema desenhado.

4.2 Aplicação Móvel

A aplicação móvel desenvolvida para o Android 4.2 (Jelly Bean), tem como principal função recolher os registos de portagens efetuados no OBE e reencaminhá-los para o *back-office* mediante iniciativa do utilizador. Além disso, uma aplicação móvel pode fornecer

funcionalidades interessantes para o âmbito do projecto, que pode passar pela criação de um ambiente gráfico onde poderão ser expostos os registos de portagem do utilizador (com data, hora, local, etc.) e dada a possibilidade de interação, onde o utilizador da aplicação poderá efetuar o pagamento das portagens incorridas.

A aplicação desenhada, nesta fase do projecto, foi baseada nas funções principais, partindo-se em primeiro lugar para uma abordagem experimental, onde se tentou estabelecer uma comunicação com o OBE, via NFC e só depois prosseguir para o desenvolvimento do ambiente de utilização. Tal como declarado nos princípios fundamentais do sistema C2S, a aplicação móvel foi arquitetada de modo a fornecer uma experiência o mais simples possível e amiga do utilizador. Para isso, encarnou-se o papel do utilizador, procurando desde cedo definir os passos de utilização com a produção de um *mockup*, representado na Figura 4.9, onde se evitou a utilização exagerada de menus e a abertura de novas *views*, ou janelas desnecessárias. A experiência de utilização fundamental, resume-se em três passos: ler os registos de portagens ("encostando" o *smartphone* ao OBE); consultar os registos de portagens; e efetuar o pagamento das pretendidas.

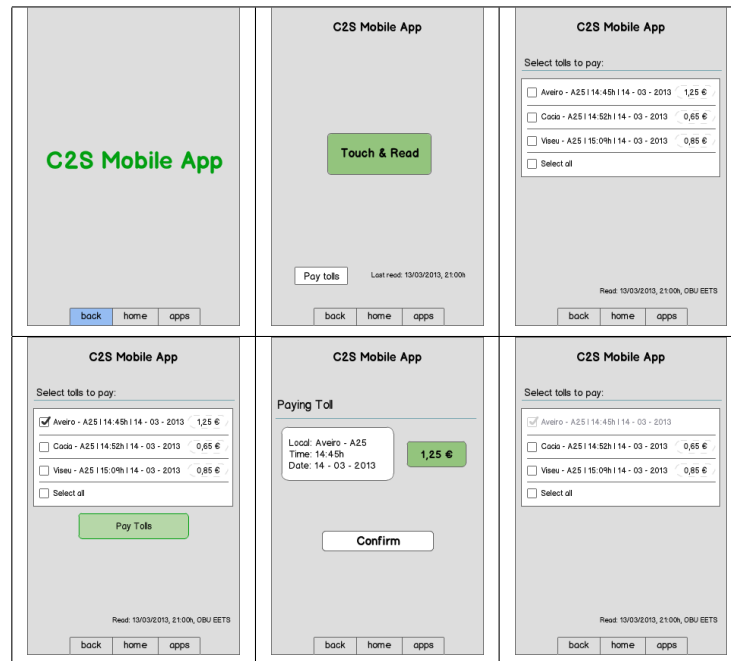


Figura 4.9: Algumas imagens do *mockup* produzido para o desenho da aplicação C2S.

Tal como se observa no *mockup*, o princípio de usabilidade da aplicação móvel assenta em três passos ler, consultar e pagar, em que os últimos dois se podem associar num só pois o último resulta de uma interação do utilizador mediante a consulta (o segundo). Portanto, desenvolveram-se duas atividades em que a primeira e principal (*main*) está responsável pelo lançamento da *view* de apresentação/introdução, quando a aplicação é lançada, e pela leitura das mensagens NFC. A segunda atividade, denominada "toll selection" possui uma lista de conteúdos, onde apresenta o registo das portagens incorridas e o respectivo campo de seleção da mesma, e lança ainda *pop-ups* de confirmação e informação, mediante a interação do utilizador. A navegação entre as duas atividades é efetuada normalmente pelo pressionar do botão "Pagar Portagens" e no caso de existirem registos, o utilizador consegue passar da atividade

principal para a segunda atividade, podendo retroceder pelo botão "back" do *smartphone*. No caso de o utilizador efetuar uma leitura, ou seja "encostar" ao OBE e se estabelecer uma comunicação com sucesso, a aplicação lança imediatamente a segunda atividade de seleção de registos.

Outro aspecto importante, reside na persistência dos dados e *logs* registados durante a operação da aplicação. Para isso criaram-se ficheiros internos com a função "openFileOutput" em modo privado ("Context.MODE_PRIVATE"), que são criados quando a aplicação é iniciada pela primeira vez e localizam-se no diretório interno da aplicação, sendo só editados no contexto da aplicação, não podendo ser acedidos (lidos ou escritos) por outras aplicações. São utilizados quatro ficheiros, dois funcionam como *buffers* de entrada, denominado "tolls toPay" e de saída, o "tolls Paid", outro ficheiro é a base de dados, o "tolls data" que possui toda a informação sobre os pórticos (ID;País;Infraestrutura;Local;Preço), e o último é o "tolls Paid receipt" que possui todos os registos pagos pelo utilizador, funciona como um recibo. Os *buffers* e o recibo são criados no *launch* da aplicação (abertura da aplicação pela primeira vez no sistema) sendo editáveis mediante as operações desencadeadas pelo utilizador, leitura dos registos NFC ou pagamento de portagens, por exemplo. Com a base de dados pretende-se um ficheiro estático não acessível pelo utilizador, apenas atualizável pelo *back-office* e com o consentimento do utilizador. Para o teste e desenvolvimento da prova de conceito, criou-se uma base de dados local a partir dos *raw asset files*, que consiste num diretório onde se pode colocar ficheiros, para serem usados no âmbito da aplicação, durante o desenvolvimento da mesma, que são posterior anexados a esta durante a sua compilação. Portanto todos os ficheiros são estáticos e se a aplicação passar para *background* ou terminar, os registos persistem, o que proporciona ao utilizador a funcionalidade de efetuar o pagamento depois de "abandonar" a viatura.

4.2.1 Implementação das atividades

No iniciar da aplicação criou-se uma pequena introdução, denominada "splash", com duração de 1 segundo, onde se poderá colocar o logotipo da aplicação, entidades parceiros e se importante, imagens com dicas e ajudas. Para isso, criou-se uma atividade extra, que é lançada quando a aplicação é iniciada, apresenta uma *view* durante 1 segundo (tempo editável) e lança a atividade principal.

A atividade principal tem como função estabelecer uma comunicação NFC com o OBE, sempre que estes se aproximem (ou "toquem"), ou seja tem como função receber os registos de portagens incorridos, disponibilizados pelo OBE e enviar os registos de portagens pagos pelo utilizador, durante a mesma troca de mensagens. Na criação desta atividade, portanto na rotina "onCreate", ilustrado na Figura 4.10, é inicializado o Intent que irá despoletar as rotinas responsáveis por atender o módulo NFC do *smartphone* e é ainda configurado o NFC através do filtro do Intent de forma a informar o sistema Android que a aplicação pretende aceder e receber notificações do módulo NFC.

De seguida, é executada a rotina "onResume" onde é lida a data e hora da última comunicação NFC estabelecida, para ser apresentada na *view* da atividade, no canto inferior direito. De seguida, é criado o botão "Pagar últimas Portagens" para o utilizador aceder à segunda atividade e efetuar pagamento de registos antigos, desde que existam registos disponíveis para pagamento, sendo esta validação efetuada pela rotina "tolls toPay available" que verifica se o ficheiro "tolls toPay" contém registos e configura a visibilidade do botão em questão. Ainda é chamada a rotina "enableNefExchangeMode" que irá ler o ficheiro "tolls_paid" para criar a

mensagem a ser enviada para o OBE, tal como descrito no esquema da Figura 4.10.

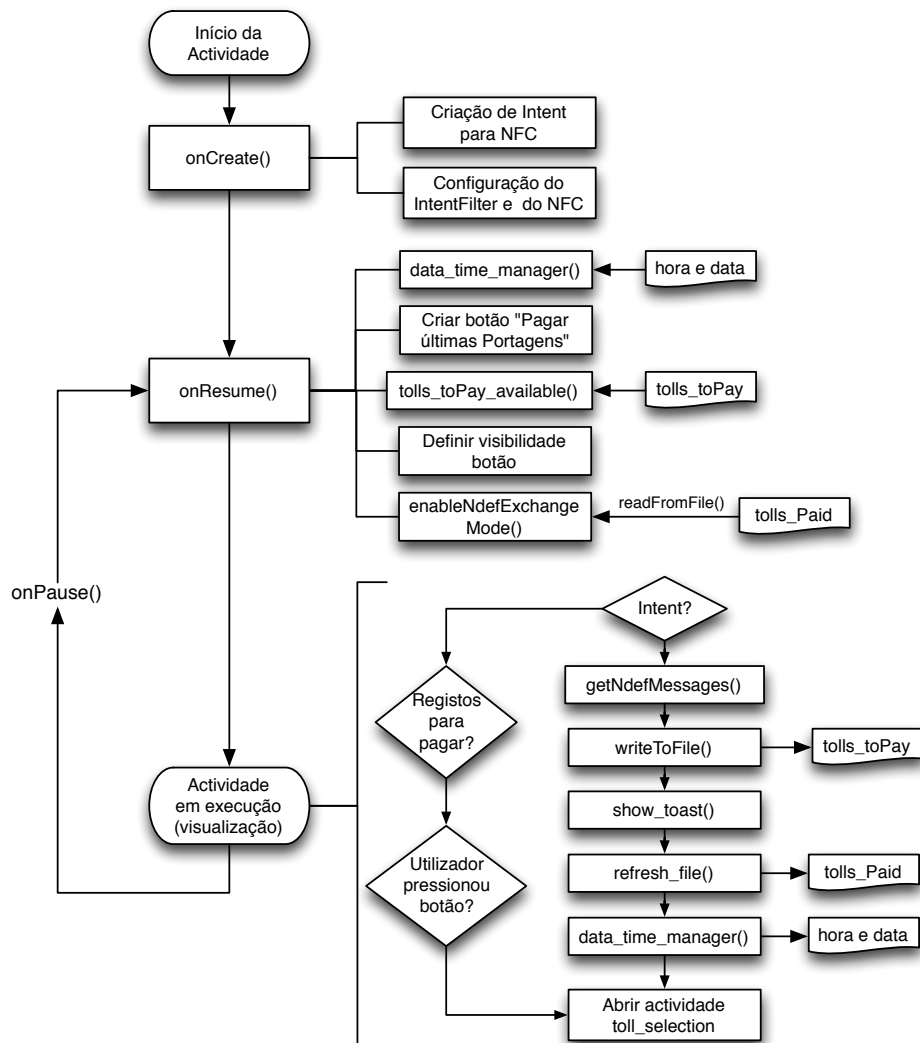


Figura 4.10: Estrutura organizacional do ciclo de vida da atividade principal ou *main*.

Durante a execução da atividade o utilizador pode premir o botão "Pagar últimas Portagens" se existirem registos de portagens a submeter para pagamento (caso em que o botão está visível) ou efetuar uma leitura do módulo NFC do OBE, e em ambos os casos a atividade "toll_selection" é iniciada.

Se o utilizador optar por ler o OBE, e contando que as comunicações NFC e o Android Beam estão ativadas nas definições do Android, será entregue uma mensagem à aplicação, o Intent, com o aviso da comunicação, sendo que é recepcionada a mensagem NDEF e enviados os registos das portagens já submetidas a pagamento. Posteriormente a rotina "getNdefMessages" interpreta a mensagem recebida e lê os registos para estruturas internas, que serão logo de seguida copiados para o ficheiro "tolls_toPay" na rotina "writeToFile". Depois destes passos, o utilizador é informado que a comunicação se estabeleceu com sucesso através de um *toast*, os registos do ficheiro "tolls_paid" são todos eliminados com a rotina "refresh_file" e a hora e data da transferência é guardada preferências partilhadas da aplicação (local onde

se pode guardar algumas variáveis que se pretendam persistentes), tal como se observa no esquema da Figura 4.10. Finalmente a atividade "toll_selection" é inicializada automaticamente depois da leitura, ficando a atividade principal em *background*, podendo futuramente ser retomada e as operações a efetuar repetirem-se a partir da rotina "onResume".

A segunda atividade, denominada "toll selection", é responsável por apresentar ao utilizador da aplicação, os registos de portagens em que incorreu e dar a possibilidade de pagamento. Na criação da atividade, com a rotina "onCreate" são inicializadas as variáveis necessárias para a atividade e carregada a *view* de fundo. Depois das configurações iniciais, a data e hora da última comunicação é carregada para ser apresentada na *view* e os registos da base de dados são lidos para registos internos, tal como se pode observar no esquema da Figura 4.11. De seguida os registos do ficheiro "tolls_toPay" são interpretados e o ficheiro "tolls_Paid_receipt" é utilizado para despistar algum tipo de erro na identificação de registos já liquidados e que ainda se mantêm no "tolls_toPay". Para se conseguir manipular os registos de forma mais flexível e eficiente, estes são transferidos dos ficheiros para *ArrayLists*.

Seguindo o ciclo de vida da atividade, ilustrado na Figura 4.11, é executada a rotina "onStart", onde é criada e configurada uma lista simples de múltipla seleção, onde são impressos os registos, para apresentar ao utilizador, produzidos pela rotina "printTolls()". Segue-se a criação e configuração do botão "Pagar" que mudará a sua visibilidade consoante a seleção de itens na lista. Para dar funções ao botão e garantir dinâmica de utilização, criam-se *listeners* que garantem que se o botão descrito for premido são apresentadas caixas de diálogo de confirmação e *toasts* (pequenas mensagens) para informar o utilizador das operações que se encontram a decorrer. Durante a execução da atividade, se o utilizador sair da aplicação ou submeter todos os registos de portagens a pagamento, (onde é lançado mais uma caixa de diálogo a informar que a lista está vazia e irá abandonar a atividade, portanto voltar a atividade principal) a atividade é reiniciada e a rotina "onStart" é novamente executada, tal como ilustra a Figura 4.11.

Tratamento e gestão de registos

Durante a execução da segunda atividade são criadas e usadas *ArrayLists*, ou seja, listas baseadas em *arrays* que disponibilizam as opções de inserção, remoção, substituição, etc. de elementos, possibilitando a utilização de estruturas de dados muito dinâmicas e flexíveis. Nesta atividade são utilizadas quatro *ArrayLists*:

1. uma lista de classes "toll", chamada "toll_data_base" para onde são lidos os registos da base de dados. É nesta lista e respectivas classes de cada elemento que assenta toda a operação de identificação e processamento de informação para o utilizador poder tomar decisões. A classe "toll", inserida em cada elemento da lista, contém alguns atributos, criados com o intuito de realizar testes e uma prova de conceito, tais como:
 - "id"(Int), que representa o ID do pórtico do registo da base de dados;
 - "country"(String), que são as iniciais do país onde se encontra o pórtico, PT por exemplo;
 - "road"(String), significa o nome da infraestrutura dos pórticos ou das zonas de portagem, A25 por exemplo;
 - "name"(String), representa o nome do local onde se localiza o pórtico, Estarreja por exemplo;

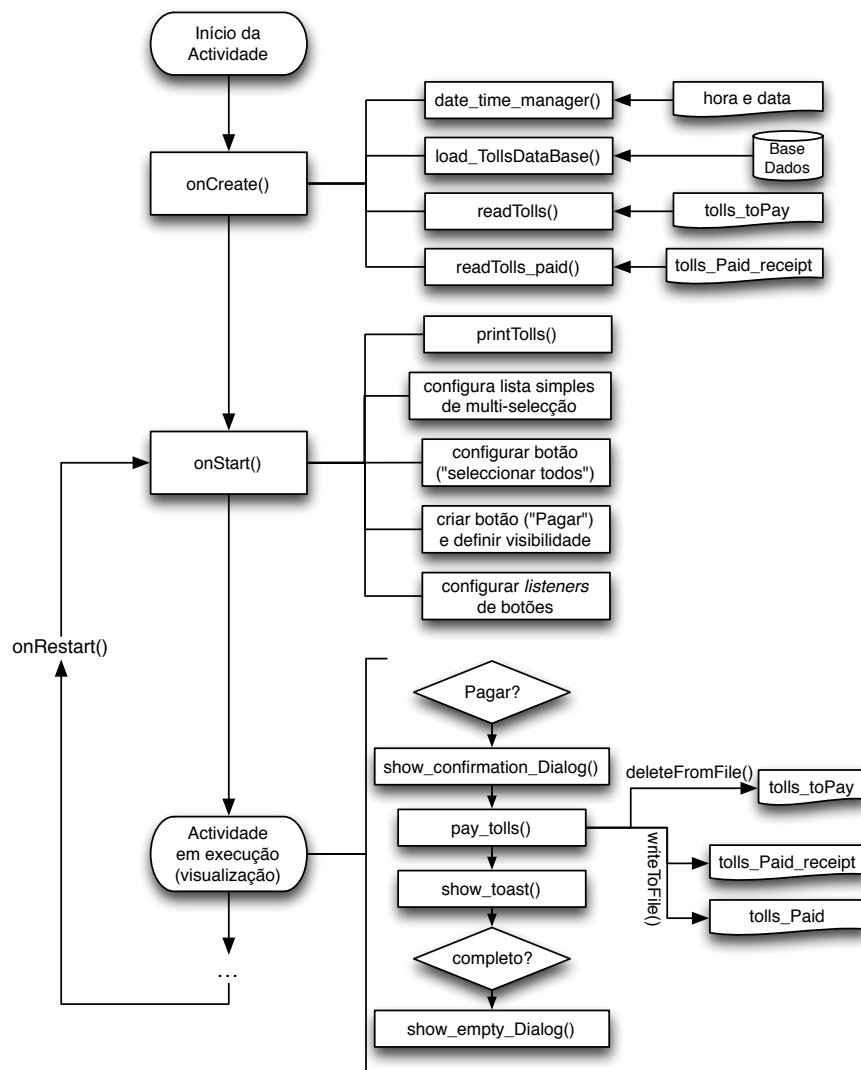


Figura 4.11: Estrutura organizacional do ciclo de vida da atividade "toll_selection".

- "price" (Double), é o preço associado à passagem de um veículo ligeiro nesse pórtico ou zona de portagem;
 - "pass_time" (ArrayList), é uma lista de subclasses "passage" que contêm os atributos "time" (String) e "id_code" (Int), servindo o primeiro para registar a data e hora em que incorreram na portagem do registo da classe "toll", e podendo ter este várias passagens, é importante existir uma sub-lista dentro desta, enquanto que o segundo atributo (o "id_code") é uma variável de apoio que define em que ordem o registo foi apresentado ao utilizador, sendo a sua utilidade explicada mais ao pormenor nos parágrafos seguintes;
2. uma lista de inteiros, denominada "toll.incurred.id", onde são registados os IDs dos pórticos que possuem registos de portagens incorridas, ou seja, é uma lista de atalhos da base de dados, sempre actualizada com os IDs (dos pórticos), onde o utilizador efetuou passagens, e que portanto são para apresentar ao utilizador. Esta lista é impor-

tante do ponto de vista organizacional e garante alguma eficiência ao sistema, visto que depois da criação da atividade, ou seja, depois da leitura dos registos incorridos, não é necessário efetuar uma nova pesquisa na base de dados (que poderá ser extensa) a cada apresentação de registos de portagens a pagar;

3. uma lista de Strings, denominada "toll_incurred_string", com o mesmo conteúdo da lista anteriormente descrita, mas desta vez com uma String informativa (Nome, Infraestrutura, Hora e Data, Preço, etc.) do pórtico ou zona de portagem em questão e que irá ser apresentada ao utilizador na lista simples de múltipla seleção;
4. por último, uma lista de Strings, de nome "tolls_paid", que armazena os registos de portagem que já foram liquidados, ou seja, a presente lista contém os elementos da lista anteriormente descrita que foram submetidos a pagamento;

Quando a atividade se inicia a base dados, definida antes da compilação nos ficheiros *asset* para a prova de conceito, é lida pela rotina "load_TollDataBase" para a lista "toll_data_base". Quando ocorre uma comunicação, entre o *smartphone* e o módulo NFC do OBE, o ficheiro "tolls_toPay" recebe os registos das portagens incorridas, ainda na atividade principal, tal como ilustra a Figura 4.12. Durante a execução da segunda atividade, o "tolls_toPay" é lido na rotina "readTolls" e nesta são preenchidos os registos dos pórticos da base de dados, por onde o cliente passou, nomeadamente a hora e data no atributo "time" da classe "passage" pertencente à lista de classes "pass_time" (onde cada elemento da lista representa uma passagem nesse pórtico). Ao mesmo tempo é preenchida a lista "toll_incurred_id" com os IDs dos pórticos onde o utilizador possui registos. De seguida é executada a rotina "readTolls_paid" que verifica junto do ficheiro "tolls_Paid_receipt" se existem registos de portagens que já foram pagos no passado mas que continuam a ser reportados no ficheiro "tolls_toPay", devido a erros de transmissão durante a comunicação NFC passada. Neste caso os registos que se comprovem já liquidados, são eliminados da lista "toll_incurred_id" e adicionados diretamente à lista "tolls_paid", como representa o esquema da Figura 4.12.

Com a lista "toll_incurred_id", a rotina "printTolls" irá criar mensagens (Strings) dos registos com os IDs da lista e guardar as mensagens produzidas na lista "tolls_incurred_string" para posteriormente serem apresentadas ao utilizador.

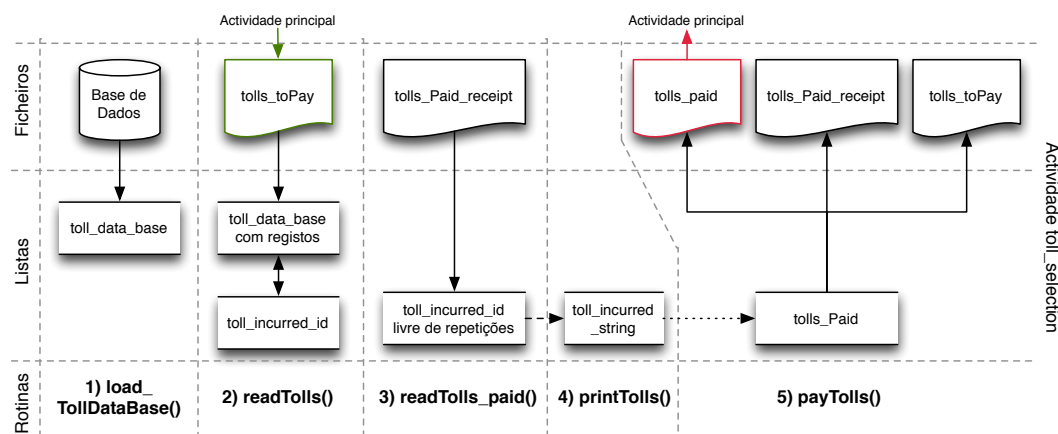


Figura 4.12: Ciclo de um registo na segunda atividade, "toll_selection".

Após um pagamento, a rotina "pay_tolls" adiciona os registos pagos na lista "tolls_paid", sendo esta lista copiada para os ficheiros "tolls_Paid" para seguir para transmissão para o módulo NFC e "tolls_Paid_receipt" para persistir no sistema como prova de pagamento local (podendo ser actualizada pelo utilizador e/ou *back-office* em versões futuras da aplicação). Ainda na rotina "pay_tolls", após o registo de todos os pagamentos, os registos de portagens são apagados do ficheiro "tolls_toPay", na rotina "delete_fromfile" e a atividade é reiniciada, sendo as listas "toll_incurred_id" e "tolls_incurred_string" automaticamente actualizadas.

Na presente fase do projecto, o campo de atuação da aplicação chega até à submissão de registos para pagamento. Para que se conclua a aplicação na totalidade é necessário desenvolver um estudo junto dos sistemas das entidades responsáveis por portagens eletrónicas, de forma a que se encontre uma ponte de conexão entre a aplicação móvel e o *back-office* do sistema, desenvolvendo por exemplo APIs nos sistemas já existentes para serem utilizadas na aplicação móvel.

4.3 Testes e Resultados

Ao longo das diversas implementações dos diferentes módulos, realizaram-se testes aos componente do sistema separadamente e começou-se a articular os módulos progressivamente, até se chegar a um sistema operacional capaz de realizar testes em cenários reais. Dado que os trabalhos começaram pelo desenvolvimento do *software* para o OBE e só depois se prosseguiu para a aplicação móvel, os respectivos testes acompanharam esta ordem. Os testes na presente fase do projecto baseiam-se maioritariamente na validação do funcionamento dos módulos e portanto numa avaliação baseada na observação. Contudo, conseguiram-se realizar algumas medições na precisão do receptor GPS e apresentar resultados da detecção de zonas de portagens. Os testes da aplicação móvel basearam-se na sua experiência de utilização e os resultados são descritos na presente secção.

4.3.1 OBE

Inicialmente testou-se a comunicação entre processos utilizando os *sockets*, de seguida procedeu-se ao teste das rotinas implementadas para aceder a ficheiros e a periféricos (como receptor GPS, botões, *buzzer* e módulo NFC). Estes testes foram desenvolvidos já no RPi conectado pela porta *ethernet* a uma rede, no laboratório, sendo os programas transferidos por Secure Copy Protocol (SCP) e editados, compilados e executados (controlados) por comunicação SSH entre o terminal (do computador) do *tester* e o OBE. Para testar as comunicações criaram-se interfaces para que o *tester* pudesse inserir e ler mensagens, observando a sua transmissão desde o *Sniffer* até à transmissão para o módulo NFC. Depois de os processos estarem conectados, começaram-se a desenvolver testes com mensagens de GPS reais e rapidamente se passou a testes de trajetos.

Para realizar os testes de trajeto, utilizou-se o programa GPS_Fake, editando-se o processo *Sniffer* para comunicar com o referido programa, de modo a simular o movimento do veículo por um trajeto, em laboratório. O GPS_Fake tem um princípio de funcionamento bastante simples e foi utilizado para substituir o receptor GPS por um "virtual". O programa cria um pseudo terminal (pseudo tty, pty) que emula um dispositivo como se fosse o receptor GPS. Este reporta tramas NMEA 0183 pela pty, que podem ser inseridas via *log file* e tanto o *gpsd* como o restante sistema a jusante interpreta os dados como se estivessem a ser reportados por um receptor GPS real. Para se criar um *log file* com as tramas NMEA 0183, utilizou-se a

plataforma Google Maps da Google Inc., pela sua versatilidade e simplicidade de utilização, para gerar um ficheiro KML (Keyhole Markup Language linguagem baseada em XML para representar posições geográficas em mapas 2D e 3D, desenvolvida pela Keyhole, Inc. adquirida pela Google Inc. [40]), sendo seguidamente convertido para o formato NMEA 0183 pela aplicação SatGenNMEA da LabSat [41]. Resumindo, com o Google Maps foi possível gerar trajetos e escolher simulações com posições reais, visto que posteriormente são convertidas para tramas NMEA 0183 e poderá-se simular diretamente no OBE, tal como ilustra a Figura 4.13. Com o *log file* produzido foi possível realizar testes de robustez do sistema, quanto à detecção de zonas de portagem como de durabilidade da execução (com ficheiros extensos e em ciclo infinito).

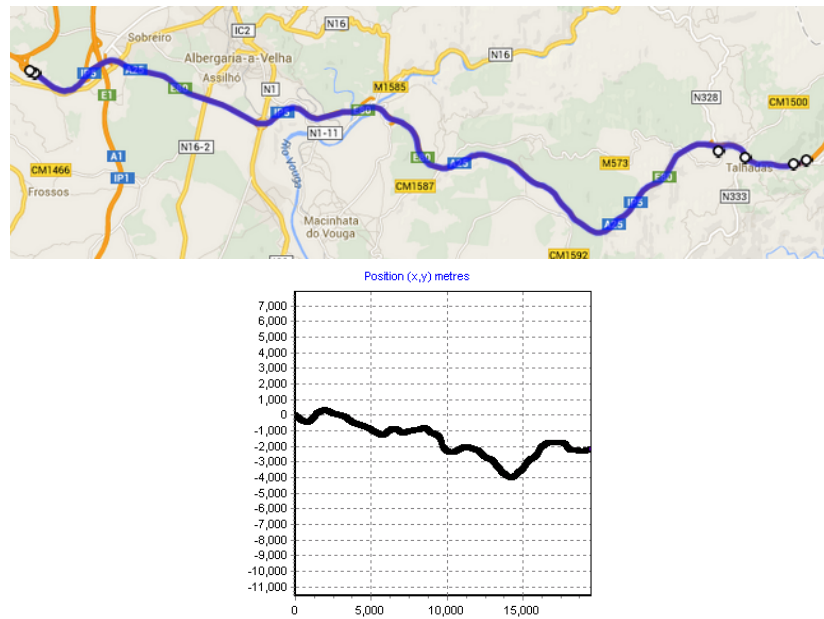


Figura 4.13: Trajeto definido e resultado do *software* SatGenNMEA.

Com este estudo foi ainda possível observar a curvatura dos traçados das autoestradas nacionais, por exemplo da A25, que ajudou a apurar o *offset* necessário para o espaçamento entre posições dos vectores, bem como o tamanho do vectores, gerados no processo *Matcher*, e já discutidos na secção anterior respectiva ao tema.

Concluído o teste de trajeto simulado, realizou-se um primeiro teste de estrada de modo a realizar-se as primeiras medições do receptor GPS e verificar se os parâmetros definidos eram apropriados para situações reais. O teste de estrada permitiu observar uma grande instabilidade nos valores decimais da latitude e longitude, a partir da quinta casa decimal (inclusive), com o veículo parado. Esta variação constante de valores deve-se a erros de recepção do receptor GPS, por onde se pode inferir que não se pode interpretar, ou retirar qualquer conclusão acima da precisão indicada (valores menores que 10^{-4}). De seguida realizaram-se os cálculos para se saber qual a distância que representa a variação de 10^{-4} graus, de forma a ter uma estimativa do erro em metros associado a este valor, e produziu-se a Tabela 4.1. Os cálculos produzidos baseiam-se no valor do raio do semi-eixo maior da Terra (localizado no equador), sendo de 6 378 137 metros segundo dados de 1989 do PT-TM06/ETRS89 -*European Terrestrial Reference System* [42], consegue-se determinar a

Ordem	Graus	0° N/S (equador) (m)	23° N/S (m)	45° N/S (m)	67° N/S (m)
0	1	111319,9491	102470,1316	78714,7668	43495,9903
1	0,1	11131,9491	10247,0132	7871,4767	4349,5990
2	0,01	1113,19491	1024,70132	787,1477	434,9599
3	0,001	111,3195	102,4701	78,7148	43,4960
4	0,0001	11,1319	10,2470	7,8715	4,3496
5	0,00001	1,1132	1,0247	0,7871	0,4350
6	0,000001	0,1113	0,1025	0,0787	0,0435
7	0,0000001	0,0111	0,0102	0,0079	0,0043

Tabela 4.1: Distância correspondente à variação decimal dos graus na Longitude para uma determinada Latitude.

distância correspondente à variação de 1 grau, dividindo o perímetro do equador (calculado pela fórmula do perímetro da circunferência) por 360° . A questão surge quando nos afastamos do equador em direção aos pólos e a latitude aumenta, diminuindo o raio da circunferência que secciona a elipse terrestre para uma determinada latitude. Neste caso o cálculo da variação da distância com o grau da longitude deverá ser calculado multiplicando a longitude pelo *coseno* da latitude, tal como ilustram os resultados das três colunas da direita da Tabela 4.1.

Observando os resultados da Tabela 4.1, pode-se concluir que as variações instáveis a partir da quinta casa decimal (inclusive), no pior dos casos (na linha do equador), acarretará um erro na ordem dos 10 metros. Portanto o *offset* escolhido para o raio da zona de portagem, denominado *range*, com o valor de 100 metros está sobredimensionado para o efeito. No entanto é importante referir que o efeito da altitude não foi tido em conta e que por vezes este pode influenciar as distâncias calculadas tornando-as maiores.

O teste de estrada foi dividido em dois ensaios, ambos em vias rápidas pertencentes à A25:

- o primeiro de curta distância (cerca de 10 km) iniciado no nó da zona industrial da cidade de Aveiro até ao nó de acesso ao porto de Aveiro, realizando-se duas viagens (ida e volta), ou seja 4 testes, onde se definiram 6 zonas de portagem, ilustrado na Figura 4.14, 3 de velocidade mais lenta (90 km/h, A, B e F) e 3 de velocidade elevada (120 km/h, C, D e E).

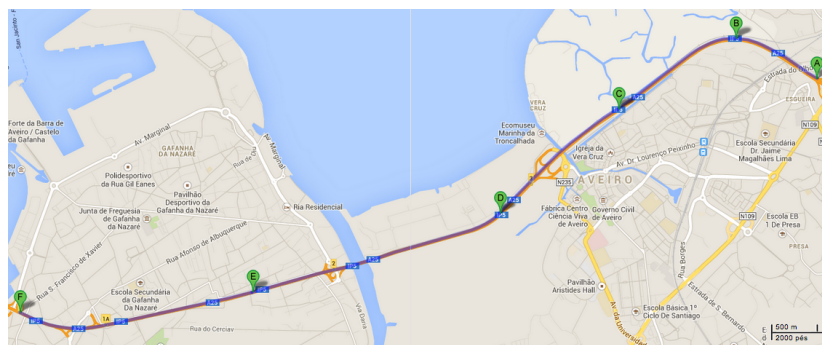


Figura 4.14: Zonas de portagem do primeiro ensaio do teste de estrada.

- o segundo ensaio, mais extenso (cerca de 85 km), ao longo da A25 entre Aveiro e Viseu, realizando-se uma viagem (ida e volta) com uma velocidade média de 120 km/h, onde se definiram 9 zonas de portagem correspondentes aos locais onde estão os pórticos físicos, ilustrados na Figura 4.15. Para se descobrir as coordenadas dos pórticos existentes foi necessário utilizar uma das viagens para se elaborar um registo das zonas, no qual foi utilizado o OBE como *data logger*.

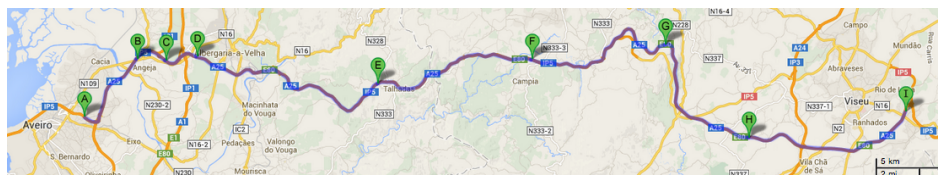


Figura 4.15: Zonas de portagem do segundo ensaio do teste de estrada, correspondentes aos pórticos existentes.

Com os testes de estrada conclui-se que as operações básicas do OBE se encontram operacionais e que as distâncias entre a posição do veículo e o centro da zona de portagem tendem a aumentar a velocidades mais elevadas ou em zonas de curvatura mais acentuadas, tal como se adivinhava na fase de desenho do sistema.

4.3.2 Aplicação móvel

Depois dos testes realizados ao OBE, seguiu-se para o desenvolvimento da aplicação móvel. Ao longo do desenvolvimento testaram-se diversos aspectos da aplicação, como o acesso a ficheiros, a persistência dos registos e o modo de utilização, de forma a esta se tornar apelativa, simples e imune a falhas. O resultado obtido foi uma aplicação organizada em duas *views* ou janelas, apresentadas alternadamente e com uma *view* de introdução, gerada no iniciar da aplicação, onde está bem patente uma experiência de utilização intuitiva, tal como demonstra a Figura 4.16.

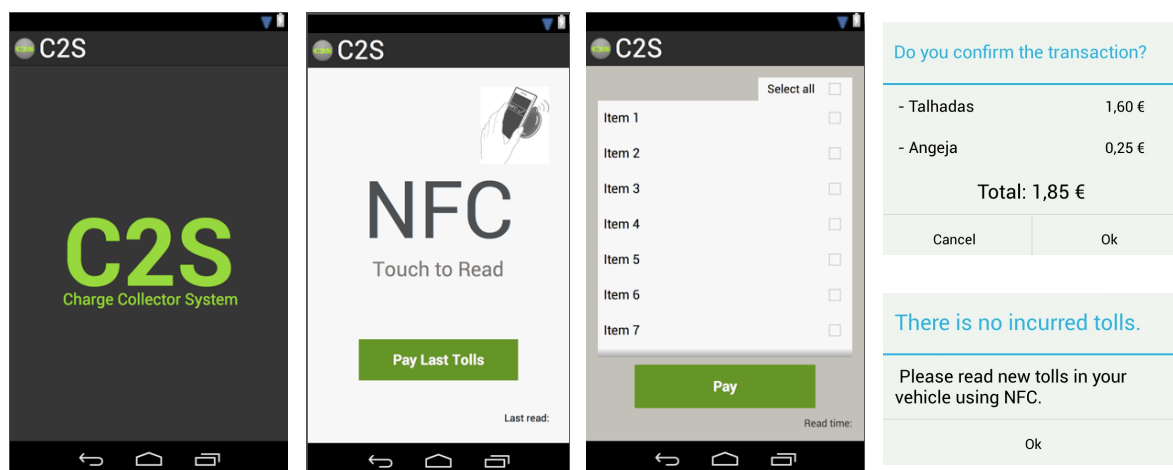


Figura 4.16: Algumas imagens da aplicação móvel C2S.

A componente mais importante da aplicação móvel é a transmissão de registos pelo módulo NFC, sendo a componente mais submetida a testes e a que apresentou maiores problemas no

seu funcionamento, realizando a operação com sucesso durante muitas vezes, mas falhando em algumas. Embora se tenham desenvolvido diferentes mecanismos de *debugging* ou depuração, como aumentar a edição e implementação de diferentes rotinas na aplicação móvel, bem como de *firmwares* no módulo do OBE, ou aumentar a tensão de alimentação dos módulos, não foi possível apurar a razão da origem deste tipo de erros, não ficando esta componente do sistema 100% operacional.

Capítulo 5

Conclusão e Trabalho Futuro

5.1 Conclusão

Com o trabalho desenvolvido no âmbito desta dissertação, é possível perceber quais os sistemas existentes em operação em cada país da União Europeia, bem como a tecnologia em que cada um se baseia e o desafio a que se propõe o EETS, os seus benefícios e as suas potencialidades. Ao mesmo tempo identificaram-se funcionalidades em falta nos sistemas atuais que limitam os utilizadores em alguns cenários, como o do *car sharing*. Estando os sistemas eletrónicos de pagamento a flexibilizarem-se, surgiu a oportunidade de inserir estas funcionalidades numa nova solução que conjugasse interoperabilidade e novas formas de utilização.

Assim, desenvolveu-se uma prova de conceito de um sistema multi-tecnologia, assente em dois módulos, um a bordo da viatura, o OBE, e outro pertencente ao utilizador, a aplicação móvel a ser executada num *smartphone*. Para o seu desenvolvimento, desenhou-se a estrutura do sistema, definiu-se a tecnologia pertencente a cada módulo, o modo como estes comunicam e partilham informação.

Numa segunda parte do trabalho, implementou-se a prova de conceito com um *single board computer* baseado em Linux, que possui como periféricos um módulo GPS e um módulo NFC, sendo este capaz de registar (efetuar *logs*) zonas de portagens, previamente definidas numa base dados local, por onde o veículo passar, e ao mesmo transmitir os referidos registos para o módulo NFC que está encarregue de comunicar com a aplicação móvel do *smartphone*.

Para efeitos demonstrativos, surgiu a necessidade de desenvolver a aplicação móvel. Então, criou-se uma aplicação para o Android 4.2, compatível com comunicações NFC, assente num princípio de utilização simples e cómodo, preparada para receber as APIs que possibilitem a conexão desta ao *back-office* de uma entidade de cobrança.

Em traços gerais, o sistema utiliza diversas tecnologias, podendo utilizar ainda mais, como por exemplo um módulo para comunicações DSRC ou comunicações móveis GPRS, que não se consideraram fundamentais nesta fase. A prova de conceito produzida representa um modelo ilustrativo do sistema, e nasce da congregação dos diversos módulos, tal como do *software* desenvolvido para cada um. Esta seguiu sempre um princípio da flexibilidade, de forma a ser facilmente compreendida e editável, mas sempre numa linha de criação de novas funcionalidades para os utilizadores.

Entre muitos desafios, esta dissertação permitiu desenvolver competências em diversas áreas, com diferentes tecnologias, tendo sido esse o seu maior desafio.

O trabalho, mais concretamente os princípios fundamentais e desenho do sistema, foram aprovados e apresentados sob artigo científico na conferência VEHICULAR 2013 em Julho de 2013, em Nice, França, na categoria de *Work in Progress*. Existe ainda um segundo artigo desenvolvido no âmbito da dissertação, já com a implementação final, que foi aceite para a conferência CETC 2013 (Conference on Electronics, Telecommunications and Computers), na qual vai ser apresentado em Dezembro de 2013, em Lisboa.

5.2 Trabalho Futuro

Ao longo dos trabalhos, dada a multidisciplinaridade do tema, nem sempre foi possível aprofundar os desenvolvimentos em determinada área em detrimento de outras, pelo que surgiram alguns pontos que devem ser explorados no seguimento deste projeto. Os referidos pontos conseguem-se correlacionar com cada bloco principal do sistema, sendo estes:

- No bloco correspondente ao do OBE, é possível desenvolver um sistema numa única placa (*single board*) que possua: um sistema computacional baseado num sistema de tempo real (*real time system*) que consiga armazenar a base de dados local, um receptor GPS *on-board* e um módulo dedicado NFC, a fim de se ganhar maior mobilidade e capacidade de teste. Para isso podem ser utilizados os algoritmos, ou/e até o código, desenvolvido se a natureza do sistema assim o permitir;
- Na aplicação móvel, seria interessante aplicar outro tipo de funcionalidades orientadas ao utilizador, como uma animação gráfica exemplificando a leitura do módulo NFC em tom de ajuda, a consulta das facturas dos pagamentos efetuados, a agregação de mais veículos e filtragem da informação por veículo, a consulta dos consumos efetuados por mês, a disponibilização de informação sobre o trânsito, o carregamento sem fios do *smartphone* e ativação do modo automóvel (kit mãos livres) quando colocado no suporte acoplado ao módulo NFC, entre outras. Para se definir um propósito, o ideal seria junto de uma entidade operadora de *car sharing* desenvolver um protótipo que integrasse as funcionalidades já utilizadas neste ambiente (como a chave do automóvel por NFC e o pagamento do aluguer via aplicação móvel) nas funcionalidades desenvolvidas no presente trabalho (registo de passagem em portagem);
- Para finalizar o protótipo, seria importante desenvolver APIs, afim de estabelecer comunicação entre a aplicação móvel e o *back-office*. O trabalho relaciona-se com a engenharia de *software* e poderia ser implementado em servidores de teste para posteriormente ser apresentado como um produto chave na mão, desta forma seria possível implementar a submissão dos registos para pagamento, a definição das opções de pagamento para o utilizador, a atualização das bases de dados e aplicação de um modelo dinâmica do valor das taxas de portagem.

Portanto, estão criadas as condições para o desenvolvimento de pelo menos 3 trabalhos baseados no projecto iniciado com esta dissertação. Por enquanto, a prova de conceito desenvolvida poderá ser integrada num sistema, a bordo de um veículo, que já utilize uma arquitetura Linux e deseje demonstrar um novo modelo de registo de portagens baseado na interação com um *smartphone*.

Apêndice A

Tabela com tecnologias dos países da União Europeia

	País	Tecnologia de pagamento baseada em	Taxa aplicada por
1	Alemanha	Vinheta ou GNSS/GSM + DSRC	Distância percorrida; Área ou zona
2	Áustria	CEN DSRC (EN15509)	Secção de estrada; Área ou zona
3	Bélgica	Vinheta, GNSS/GSM+ DSRC em 2013 [1]	Área ou zona
4	Bulgária	Vinheta [2]	Licença de utilização
5	Chipre	-	-
6	Dinamarca	CEN DSRC	Área ou zona
7	Eslovénia	CEN DSRC (EN15509)	Secção de estrada
8	Eslováquia	GNSS/GSM + CEN DSRC	Distância percorrida
9	Espanha	CEN DSRC (PISTA + EN15509)	Secção de estrada; Área ou zona
10	Estónia	-	-
11	Finlândia	-	-
12	França	CEN DSRC (TIS + EN15509) Introduzindo GNSS/GSM + CEN DSRC	Secção de estrada; Área ou zona
13	Grécia	CEN DSRC (A1)	Secção de estrada; Área ou zona
14	Hungria	Vinheta electrónica [3]	Licença de utilização
15	Irlanda	CEN DSRC (A1)	Secção de estrada; Área ou zona
16	Itália	UNI DSRC 10607	Secção de estrada; Área ou zona
17	Letónia	Manual	Área ou zona
18	Lituânia	Manual	Área ou zona
19	Luxemburgo	-	-
20	Malta	ANPR	Área ou zona
21	Países Baixos	Manual	Área ou zona
22	Polónia	CEN DSRC (EN15509)	Secção de estrada
23	Portugal	CEN DSRC (EN15509)	Secção de estrada; Área ou zona
24	Reino Unido	CEN DSRC (EN15509)	Secção de estrada
25	República Checa	CEN DSRC (EN15509)	Secção de estrada
26	Roménia	Vinheta	Licença de utilização
27	Suécia	CEN DSRC (EN15509)	Área ou zona

Apêndice B

Comparação de OBUs com tecnologia GNSS-GSM e DSRC

On-Board Unit	Continental AG				Fela Management AG		Kapsch
	Obu 1372 Dashboard Solution	Obu 1373 DIN Slot Solution	Obu 1374 Windscreen Solution	Obu 1376 The Swiss LSVA	TRIPON-CH	TRIPON-EU	TRIPON-SM
Fabricante							
Comunicações	GPS DSRC GSM/GPRS	GPS GSM/GPRS	GPS DSRC GSM/GPRS	DSRC Bluetooth	GNSS-CEN* DSRC TC278** GSM/GPRS	GNSS-CEN* DSRC TC278** GSM/GPRS	GNSS-CEN* DSRC TC278** GSM/GPRS
Periféricos	Leitor de smart card externo Giroscópio Interface CAN	Leitor de smart card externo Giroscópio Interface CAN	Leitor de smart card externo GSM/GPRS	Leitor de smart card externo Interface CAN	Giroscópio Interface CAN GPIO para tacógrafo, ignição, sensor de reboque	Interface CAN GPIO para tacógrafo, ignição, sensor de reboque	Módulo de comunicações TS3220/20A Ligação a fonte de energia (ao isqueiro)
Fonte de Alimentação	Do veículo (não há informação)	Do veículo (não há informação)	(não há informação)	(não há informação)	Fonte do veículo 9 – 50V DC	Fonte do veículo 9 – 50V DC	Fonte do veículo 8 – 48V DC
Dimensões e Peso	(informação indisponível)	(informação indisponível)	(informação indisponível)	(informação indisponível)	155x170x80mm 600g	185x55x172mm 950g; 135x55x172mm 190g (Módulo de comunicações)	128x68x32mm 127g (excluindo o módulo de comunicações)
Condições Ambientais	(informação indisponível)	(informação indisponível)	(informação indisponível)	(informação indisponível)	-25°C a +85°C 10% a 90% de humidade 50G de choque durante 11ms EN60068	-25°C a +85°C 10% a 90% de humidade 50G de choque durante 11ms EN60068	-25°C a +85°C EN 60721

Bibliografia

- [1] Gabriel Nowacki. Iteroperabilty problems of especially electronic toll collection. pages 4–17, 2011.
- [2] Transdev Mobilidade SA. Citizenn car-sharing, 2009.
- [3] Getaround Inc. Getaround - car sharing, 2009.
- [4] Continental AG. Tolling On-Board Units, 2013.
- [5] Pedro Pinto. The experience of Ascendi in design and implementing a Multi-Lane Free-Flow Tolling System (MLFF). In *IRF World Meeting 2010 – 16 th World Meeting*, number Lim, pages 17–24, Lisboa, 2010.
- [6] IBM. Socket programming, 2013.
- [7] K Persad, CM Walton, and S Hussain. Toll Collection Technology and Best Practices Vehicle. pages 1–2, 2007.
- [8] Parlamento Europeu. Assembleia da república. pages 5030–5032, 2007.
- [9] European Parliament; European Council. Directive 2004/52/EC interoperability of electronic road toll systems in the Community, 2011.
- [10] Ministério da Economia e do Emprego. Diário da República. pages 5094–5100, 2011.
- [11] Ascendi. Registo nacional do Sistema Electrónico Europeu de Portagem (SEEP) em Portugal. 2012.
- [12] Toll Collection and Technology Overview. Toll Technology Considerations , Opportunities , and Risks. 2:1–33, 2006.
- [13] AG Andreas, Weiss, EFKON. General Department of National Roads and Motorways. 2009.
- [14] IBTTA. IBTTA 2011 Annual Meeting, 2011.
- [15] K Persad, CM Walton, and S Hussain. Electronic vehicle identification: Industry standards, performance, and privacy issues. pages 1–3, 2007.
- [16] Nations Online. Union European Members.
- [17] Stephanie Steinberg and Bill Vlasic. Car-Sharing Services Grow, and Expand Options, 2013.

- [18] William Vickrey. Principles of Efficient Congestion Pricing, 1992.
- [19] N Velaga, Kate Pangbourne, and K Papangelis. GNSS-based Dynamic Road User Charging System. pages 11–13, 2010.
- [20] ITS International. Debating contactless toll charging by smartphone, 2012.
- [21] GeoToll. GeoToll, 2012.
- [22] FELA Management AG. Tripon CH. 2001.
- [23] FELA Management AG. Tripon - EU OBU - The all in one European OBU Solution for Windschild Installation.
- [24] FELA Management AG. Tripon - EU DIN OBU - Interoperable OBU Solution for Europe.
- [25] FELA Management AG. Tripon - SM OBU - The next step in GNSS based tolling.
- [26] Kapsch Group. On Board Unit TS3209. pages 9–10.
- [27] National Coordination Office for Space-Based Positioning, Navigation. GPS space segment, 2013.
- [28] ELinux wiki. RPi Distributions, 2013.
- [29] Debian.org. Versões/Lançamentos Debian, 2013.
- [30] Chris Tyler. Raspberry Pi Fedora Remix, 2012.
- [31] RISCOS Ltd. RISC OS, 2013.
- [32] Nancy Gibbs. Your Life Is Fully Mobile. *Time Tech*, MOBILE TEC, 2012.
- [33] Eric Pfanner. Smartphone Rivals Gain On Samsung And Apple. *Times, New York*, New York e:B1, 2013.
- [34] NFC World. List of NFC - enabled mobile devices, 2013.
- [35] USGlobalSat Inc. USGlobalsat BU 353, 2012.
- [36] Marine Electronic Devices. National Marine Electronics Association (US). *NMEA 0183-Standard for Interfacing Marine Electronic Devices*, 2002.
- [37] CATB. CATB, 2013.
- [38] Arduino. Arduino Uno, 2013.
- [39] Seeed Studio. NFC Shield V1.0, 2013.
- [40] Google Inc. Google Acquires Keyhole Corp, 2004.
- [41] LabSat Racelogic Ltd. SatGen NMEA. Technical report, 2013.
- [42] Associação Internacional de Geodesia. European Terrestrial Reference System, 1997.